BAB III
METODE PENELITIAN

Tahapan metodologi yang akan dilakukan pada penelitian ini menggunakan *Cross Industry Standard Process Model for Data Mining* (CRISP-DM). Berikut tahapan pada CRISP-DM.

3.1 *Business Understanding*

Pemahaman masalah penelitian mengacu pada segmentasi mahasiswa yang kesulitan menentukan topik karena keterbatasan pustaka, kemudian dalam mencari daftar pustaka dari penelitian sebelumnya mahasiswa harus memilih satu per satu secara manual di perpustakaan. Berdasarkan masalah tersebut perlu dilakukan metode khusus agar topik tugas akhir dan skripsi dapat dikelompokkan.

3.1.1 Tujuan Bisnis

Penelitian ini dilakukan untuk membantu mahasiswa agar tidak kesulitan menentukan topik karena keterbatasan pustaka dan agar mahasiswa tidak mencari satu per satu daftar pustaka secara manual di perpustakaan.

3.1.2 Situasi Bisnis

Situasi yang terjadi pada penelitian ini adalah semua skripsi dan tugas akhir program studi Informatika masih bercampur, tanpa ada pembeda antara topik satu dengan topik lainnya.

3.1.3 Tujuan *Data Mining*

Tujuan dari data mining adalah mengelompokkan topik tugas akhir dan skripsi program studi Informatika Universitas Sahid Surakarta.

3.1.4 Perencanaan Strategi

Strategi yang dilakukan adalah dengan mengajukan penelitian ke perpustakaan Universitas Sahid Surakarta, kemudian pengumpulan data dari perpustakaan, pengolahan data, terakhir melakukan analisa hasil.

3.2 *Data Understanding*

Pada tahap ini dilakukan pengumpulan data awal dengan wawancara dan observasi.

3.2.1 Wawancara

Wawancara dilakukan dengan petugas perpustakaan Universitas Sahid Surakarta, dengan mengajukan pertanyaan seputar kajian yang akan diteliti sebagai landasan dalam memperkuat peneliti dan untuk menjawab permasalahan. Wawancara ini digunakan untuk mendapatkan data skripsi dan tugas akhir mahasiswa prodi Informatika Universitas Sahid Surakarta periode wisuda ke-21 sampai ke-28.

3.2.2 Observasi

Data diambil langsung pada perpustakaan Universitas Sahid Surakarta melalui petugas perpustakaan. Data yang diminta adalah tugas akhir dan skripsi prodi informatika Universitas Sahid Surakarta periode wisuda ke-21 sampai ke-28 dengan abstrak berbahasa Indonesia. Data yang didapat sebanyak 143.

3.3 *Data Preparation*

Data yang telah didapatkan diolah terlebih dahulu dengan menyiapkan data. Ada 2 tahap dalam proses persiapan data :

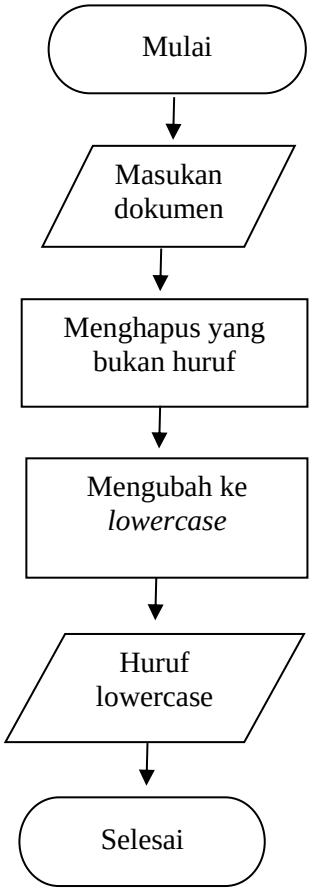
3.3.1 *Preprocessing data*

Data abstrak tugas akhir dan skripsi mahasiswa diolah dalam proses *preprocessing* dengan tujuan untuk memastikam data yang akan diolah pada proses selanjutnya adalah data yang baik. *Preprocessing* terdiri dari lima proses, yaitu *tokenization*, *case folding*, *filtering*, *normalisasi*, *stemming*, dan *stopword*. Berikut penjelasan dari masing-masing proses:

a. *Case folding*

Menghapus karakter yang bukan merupakan huruf, seperti tanda baca dan angka. Kemudian mengubah seluruh teks menjadi *lowercase.*

Detail proses *case folding* dapat dilihat pada Gambar 3.1.



Gambar 3.1 Bagan alir *Case folding*

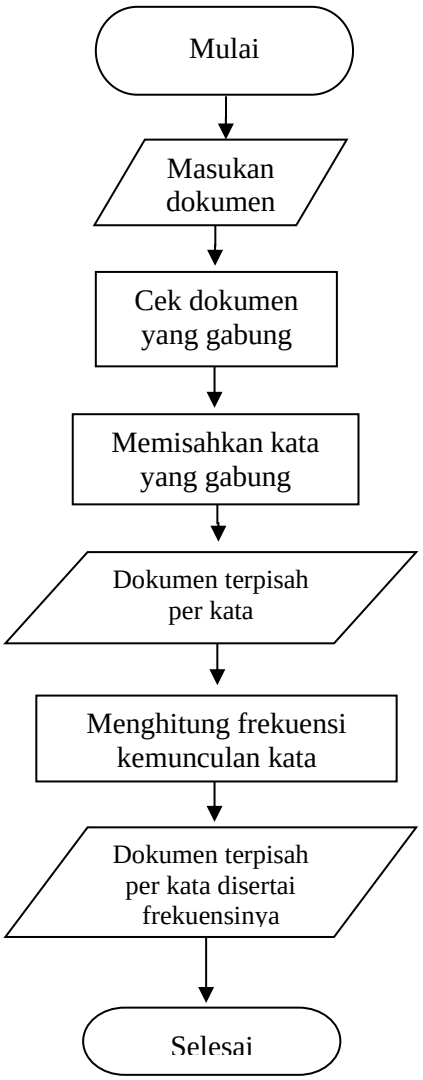
Hasil dari penerapan *case folding* dapat dilihat pada Tabel 3.1.

Tabel 3.1 Contoh Penerapan *Case folding*

Sebelum	Sesudah
Proses bimbingan skripsi di Universitas Sahid Surakarta (USAHID) yaitu dengan datang ke kampus untuk menemui dosen pembimbing secara langsung, sedangkan tidak setiap waktu mahasiswa dengan dosen pembimbing dapat bertemu dikarenakan dosen pembimbing	proses bimbingan skripsi di universitas sahid surakarta usahid yaitu dengan datang ke kampus untuk menemui dosen pembimbing secara langsung sedangkan tidak setiap waktu mahasiswa dengan dosen pembimbing dapat bertemu dikarenakan dosen pembimbing

b. *Tokenization*

Pada tahap ini *string* dokumen akan dipecah per kata berdasarkan spasi dan tanda penghubung (-). Pada tahap ini dokumen yang awalnya merupakan satu *string* panjang akan dipenggal pada setiap kata, sehingga menjadi banyak string kemudian kata tersebut akan dicari frekuensi kemunculan kata. Berikut merupakan alur dari tahap *tokenization* digambarkan pada Gambar 3.2.



Gambar 3.2 Bagan alir *Tokenization*

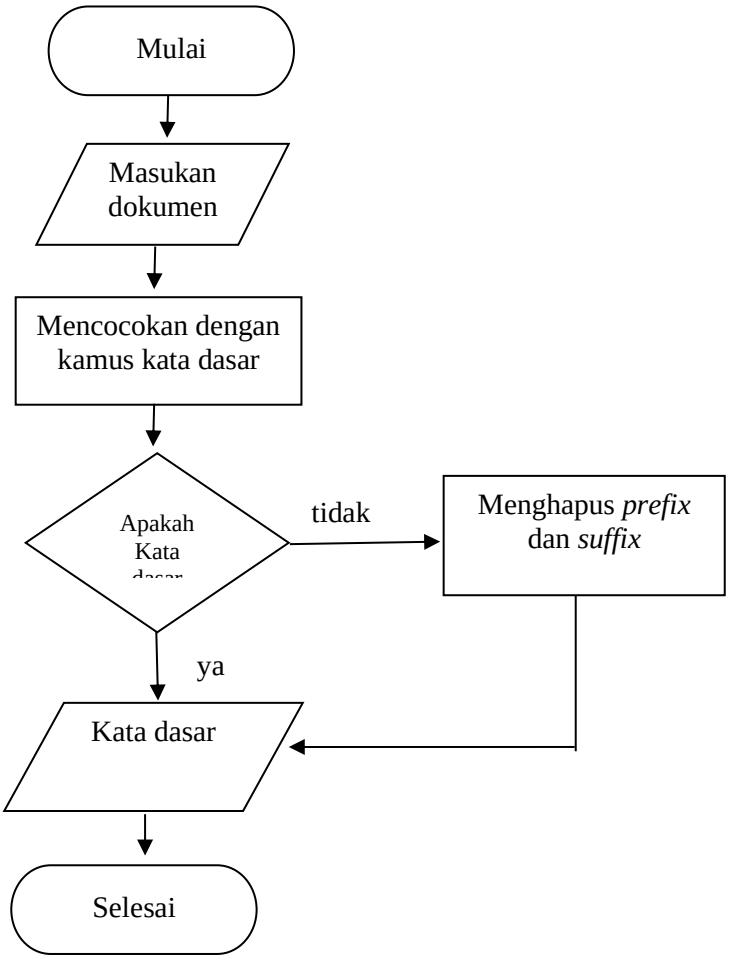
Hasil dari penerapan *tokenization* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Contoh Penerapan *Tokenization*

Sebelum	Sesudah
Proses bimbingan skripsi di Universitas Sahid Surakarta yaitu dengan datang ke kampus	Proses = 1 Bimbingan = 1 Skripsi = 1 Di = 1 Universitas = 1 Sahid = 1 Surakarta = 1 Yaitu = 1 Dengan = 1 Datang = 1 Ke = 1 Kampus = 1

c. *Stemming*

Proses ini akan mencocokkan satu per satu kata pada dokumen ke kamus kata dasar. Bila kata tersebut ada dalam kamus kata dasar, maka kata tersebut dianggap sebagai kata dasar. Bila kata tersebut tidak ada, maka akan dilakukan penghapusan *suffix* dan *prefix* dan kata tersebut akan dicek kembali ke kamus kata dasar kemudian dianggap kata dasar bila ada dalam kamus. Misalnya kata “menunjukkan”, “ditunjukkan” akan ditransformasi menjadi kata “tunjuk”. Detail proses *stemming* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Bagan alir *Stemming*

Hasil dari penerapan *stemming* dapat dilihat pada Tabel 3.3.

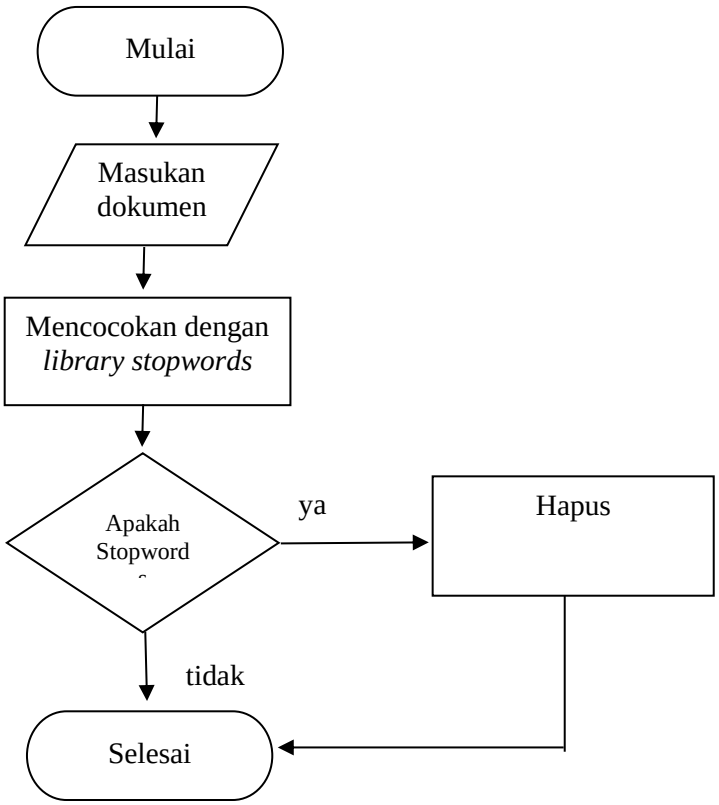
Tabel 3.3 Contoh penerapan *stemming*

Sebelum	Sesudah
Proses bimbingan skripsi di Universitas Sahid Surakarta yaitu dengan datang ke kampus untuk menemui dosen pembimbing secara langsung	Proses bimbing skripsi di universitas sahid surakarta yaitu dengan datang ke kampus untuk temu dosen bimbing cara langsung

d. *Filtering*

Proses ini dilakukan dengan membuat library *stopwords* yang berisi kata hubung yang akan dihapus, kemudian mencocokkan seluruh kata pada

dokumen ke library *stopword* tersebut, apabila ada kata yang masuk ke library *stopwords*, maka kata tersebut akan dihilangkan dari dokumen. Contoh *stopword* dalam Bahasa Indonesia adalah “yang”, “dan”, “di”, “dari”, dan lain-lain. Makna di balik penggunaan *stopword* yaitu dengan menghapus kata-kata yang memiliki informasi rendah dari sebuah teks, sehingga kita dapat fokus pada kata-kata penting. Bagan alir proses *filtering* dapat dilihat pada Gambar 3.4.



Gambar 3.4 Bagan alir *Filtering*

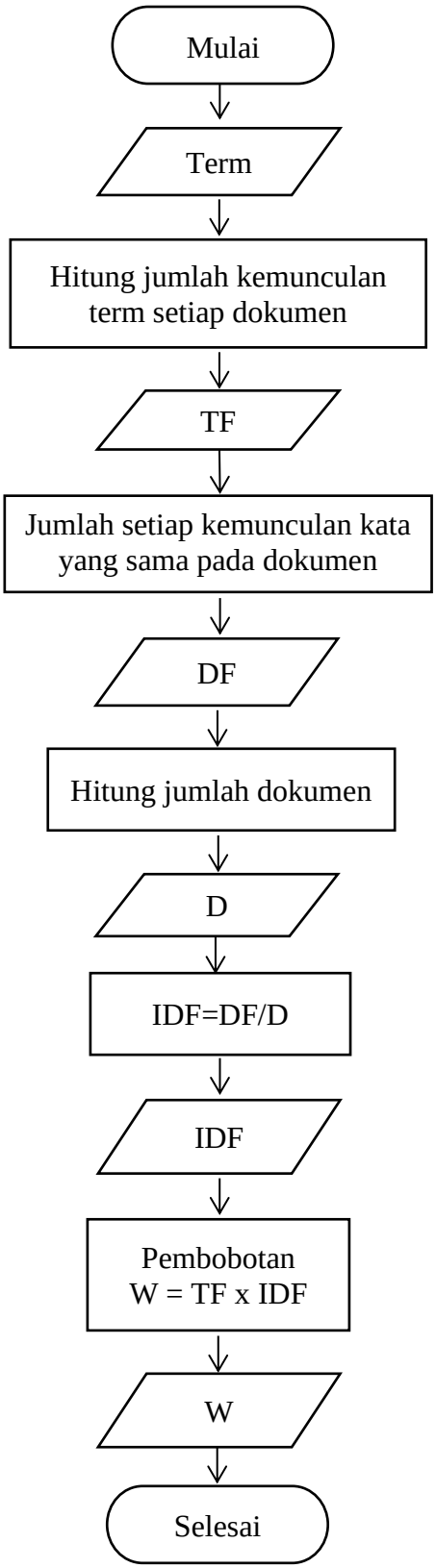
Hasil dari penerapan *stemming* dapat dilihat pada Tabel 3.4.

Tabel 3.4 Contoh Penerapan *filtering*

Sebelum	Sesudah
Proses bimbingan skripsi di Universitas Sahid Surakarta yaitu dengan datang ke kampus untuk menemui dosen pembimbing	Proses bimbingan skripsi universitas sahid surakarta datang kampus menemui dosen pembimbing

3.3.2 TF-IDF (*Term Frequency-Inverse Document Frequency*)

Perubahan data dilakukan dengan melakukan pembobotan kata (*term*). Pembobotan *term* digambarkan dalam sebuah bagan alir pada Gambar 3.5.



Gambar 3.5 Bagan alir TF-IDF

Metode TF-IDF ini akan menghitung nilai TF (*Term Frequency*) dan nilai IDF (*Inverse Document Frequency*) pada setiap *term*.

Misalkan terdapat sebuah term “sistem” yang muncul sebanyak 5 kali dalam sebuah dokumen, sedangkan kemunculan term terbanyak dalam dokumen tersebut sebanyak 10 kali, Maka nilai tf “sistem” adalah 0,5. Dengan *proses perhitungan sebagai berikut:*

$$^{5}/_{10} = 0{,}5$$

Nilai DF *term* “sistem” adalah 15 dari total dokumen 144 maka niali IDF adalah

$$\log \frac{143}{15} = 0{,}98$$

Dari hasil diatas maka dapat diketahui nilai TF-IDF dari term “sistem” adalah 0,45. Dengan proses perhitungan sebagai berikut:

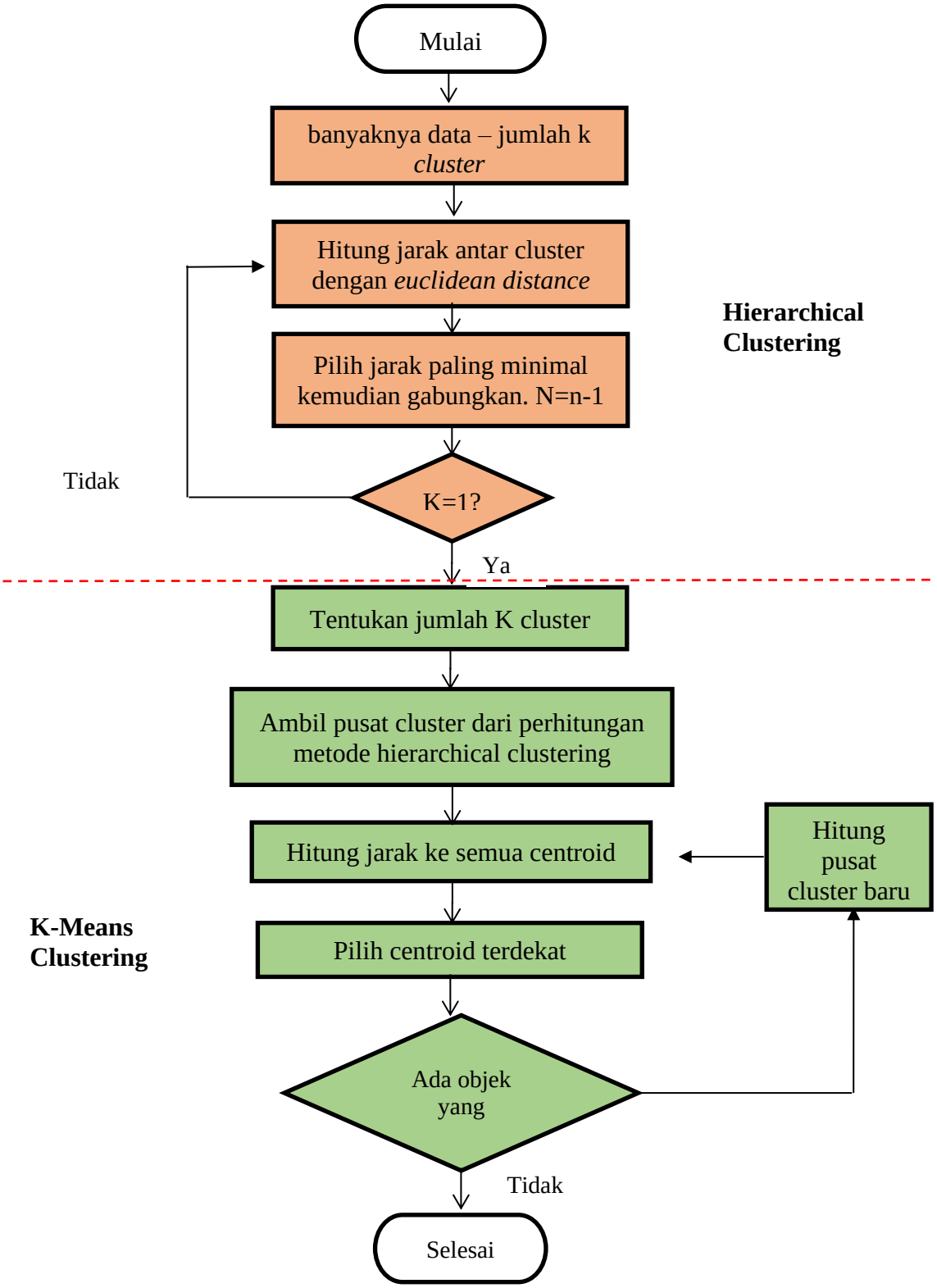
$$0{,}5x0{,}9 = 0{,}45$$

Hasil pembobotan *term* digunakan untuk menghitung bobot setiap dokumen dengan cara menjumlahkan semua bobot *term* pada suatu dokumen.

Perhitungan bobot dokumen dilakukan secara bertahap dimulai dengan menghitung TF, DF, IDF, dan TF-IDF. Proses perhitungan bobot ini dilakukan pada semua dokumen sebanyak 143.

3.4 *Modelling*

Proses *clustering* terdiri dari dua proses. Proses pertama adalah proses *hierarchical clustering*. Sebelum dilakukan pengelompokkan, setiap data yang ada akan dianggap sebagai *cluster*. Bagan alir untuk proses *clustering* dapat dilihat pada Gambar 3.6.



Gambar 3.6 Bagan algoritma *Hierarchical clustering* dan *K-means clustering*

Apabila terdapat jumlah data sebanyak n, dan k dianggap sebagai jumlah *cluster*, sehingga besarnya n adalah sama dengan k (n=k). Selanjutnya, penelitian ini menggunakan *Euclidean Distance Space* untuk menghitung jarak antar *cluster* berdasarkan jarak rata-rata antar objek.

Berdasarkan dari hasil perhitungan tersebut pilih jarak yang paling minimal kemudian gabungkan, maka besarnya n adalah n-1 (n = n - 1). Jarak *cluster* akan di-*update* ketika 2 *cluster* digabungkan. Setelah proses ini selesai, maka akan dihasilkan sebuah dendogram, dendogram menggambarkan proses penggabungan *cluster* sehingga menjadi *cluster* yang lebih tinggi.

Contoh menentukan jarak *euclideaan distance* dari 3 data yang menggunakan 2 variable :

data	X	Y
1	1	1
2	4	1
3	1	2

Penyelesaian :

$$d_{1.1}(data\ 1\ , data\ 1) = |1-1| + |1-1| = 0$$
$$d_{2.1}(data\ 2\ , data\ 1) = |4-1| + |1-1| = 3$$
$$d_{3.1}(data\ 3\ , data\ 1) = |1-1| + |2-1| = 1$$
$$d_{2.2}(data\ 2\ , data\ 2) = |4-4| + |1-1| = 0$$
$$d_{2.3}(data\ 2\ , data\ 3) = |4-1| + |2-1| = 4$$
$$d_{3.3}(data\ 3\ , data\ 3) = |1-1| + |2-2| = 0$$

Data	1	2	3
1	0	3	1
2	3	0	4
3	1	4	0

Setelah jarak data teridentifikasi langkah selanjutnya adalah menggabungkan data yang memiliki jarak terdekat menggunakan metode *single linkage*. Dari matrik diatas jarak terdekat ada pada data ke-1 dan ke-3 dengan nilai 1. Hitung antara jarak gabungan antara jarak minimal terpilih dengan masing-masing data yang belum tergabung.

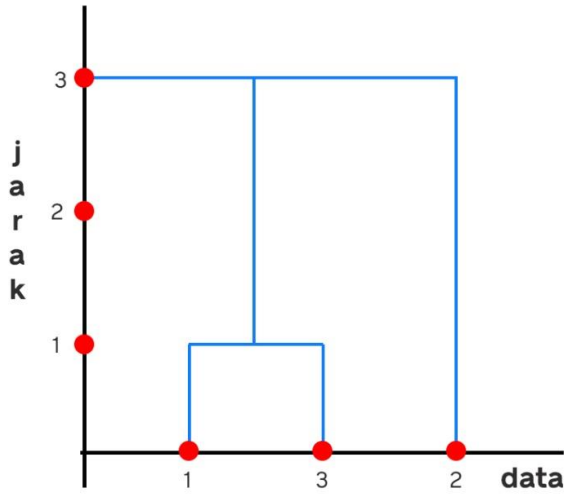
Rumus *single linkage* :

$$d_{xy} = \min\{d_{xy}\}, d_{xy} \in D$$

$$d_{(1,3)(2)} = \min?\,(d_{1.2}, d_{3.2} = \min\{3,4\} = 3$$

Data	1,3	2
1,3	0	3
2	3	0

Terdapat dua cluster yang diperoleh, cluster satu dengan anggota data 1 dan 3, kemudian cluster dua dengan angota 2. Jarak dari dua cluster ini sama yaitu 3, maka akan digabung menjadi satu cluster D(1,2,3). Berikut adalah *dendogram* yang terbentuk dapat dilihat pada Gambar 3.7.



Gambar 3.7 Dendogram *sample data*

K-means clustering akan dijalankan dengan hasil dari *hierarchical clustering* sebagai tittik awal. Metode *k-means* akan mengoptimalkan posisi *centroid* dengan melakukan hitungan berulang pada *centroid* dari tiap *cluster*. Penghitungan ini akan terus berlangsung hingga nilai *centroid* stabil atau batas iterasi tercapai. Setelah *k-means* mencapai *centroid* yang stabil, maka nilai *centroid* dianggap sudah akurat.

3.5 *Evaluation*

Tahap ini dilakukan analisis hasil *clustering* dengan cara menghitung *average intra similarity* dan *average inter similarity* dari setiap *cluster*. Dengan mengamati

nilai *similarity* tersebut, dapat dianalisa kualitas dari *cluster* yang terbentuk pada tahap sebelumnya. Nilai *average intra similarity* mencerminkan similaritas dari setiap dokumen yang ada pada satu *cluster*, bila nilai *average intra similarity* tinggi, maka *cluster* dianggap baik karena setiap dokumen yang ada dalam *cluster* memiliki *similarity* yang tinggi. Nilai *average inter similarity* mencerminkan jarak antara satu *cluster* dengan *cluster* lainnya, semakin kecil nilai *average inter similarity*, maka semakin baik pula *cluster* tersebut karena *cluster* tersebut terpisah dari *cluster* lainnya.

3.6 *Deployment*

Pada tahap ini, informasi yang telah diperoleh akan dipresentasikan dalam bentuk laporan. Hasil penelitian ini berupa pengelompokkan skripsi dan tugas akhir mahasiswa prodi Informatika Universitas Sahid Surakarta.