ANALISIS PERBANDINGAN CLUSTERING-BASED, DISTANCE-BASED DAN DENSITY-BASED DALAM MENDETEKSI OUTLIER

Dedy Handriyadi [1], M.Arif Bijaksana, Ir .MTech [1,] Erwin Budi Setiawan, MT[2]
[1]*Jurusan Teknik Informatika, Fakultas Teknik Informatika, IT Telkom Bandung*
[2]*Jurusan Ilmu Komunikasi, Fakultas Sains, IT Telkom Bandung*
E-mail: ndogBosok@yahoo.com, mab@ittelkom.ac.id, erw@ittelkom.ac.id

ABSTRAK

Data Mining adalah proses pencarian pola-pola dan kecenderungan yang menarik dari dalam basis data berukuran besar. Sebuah outlier didefinisikan sebagai sebuah titik data pada suatu data set dimana sangat berbeda dibandingkan dengan titik data pada data set pada umumnya dengan suatu ukuran tertentu. Outlier ini walaupun mempunyai kelakuan yang abnormal, seringkali mengandung informasi yang sangat berguna. Permasalahan deteksi outlier ini mempunyai peran yang sangat penting pada aplikasi deteksi kecurangan, analisis kekuatan jaringan dan deteksi intrusi. Pencarian outlier biasanya dengan konsep keterdekatan berdasarkan hubungannya dengan sisa data yang ada. Pada data berdimensi tinggi, kepadatan data akan semakin berkurang, akibatnya dugaan akan keterdekatan antar data menjadi gagal.Pada makalah ini akan dilakukan perbandingan metode dalam pencarian suatu outlier dalam data berdimensi tinggi. Metode yang akan dibandingkan yaitu: Clustering-based, Distance-based, dan Density-based. Dimana masing-masing metode telah mendukung data berdimensi tinggi.

***Kata Kunci** : data mining, outlier, deteksi outlier, metode deteksi outlier.*

1. PENDAHULUAN

1.1 Latar Belakang

Dewasa ini ledakan data hampir terjadi di setiap penjuru dunia baik industri, instansi dan internet. Dengan kondisi seperti ini terdapat banyak tuntutan untuk menemukan informasi berguna yang tenggelam dalam tumpukan data dari berbagai sumber. Data dengan jumlah yang begitu besar ini akan sangat menyulitkan apabila kita ingin menganalisa apakah terdapat suatu kesalahan dalam data tersebut. Data yang mempunyai sifat dan karakteristik yang berbeda dari data – data pada umumnya dan mempunyai kemunculan kejadian relatif sedikit dikatakan sebagai outlier.

Sebuah outlier dapat didefinisikan sebagai sebuah titik data pada suatu database dimana sangat berbeda dibandingkan dengan titik data pada database pada umumnya dengan suatu ukuran tertentu. Titik ini seringkali mempunyai informasi yang sangat berguna yang didefinisikan data pada kelakuan sistem yang abnormal. Teknik deteksi outlier digunakan pada aplikasi kecurangan kartu kredit, network intrusion detection, aplikasi keuangan dan lain lain.

Banyak metode data mining dalam pencarian outlier seperti clustering yang mendefinisikan sebuah outlier tidak terdapat dalam cluster tersebut, dengan kata lain, clustering secara implisit mendefinisikan outlier sebagai noise dari suatu cluster tertentu. Teknik lainnya mendefinisikan outlier sebagai titik dimana bukan dari bagian cluster maupun noise cluster tersebut, akan tetapi titik tertentu yang berkelakuan sangat berbeda dengan keadaan yang normal. Metode statistik dengan medefinisikan sebuah outlier berada diluar sekumpulan data yang ada. Metode distance-based mendefinisikan sebuah outlier berada jauh dari pusat data. Metode density-based mendefinisikan sebuah outlier merupakan sekumpulan titik data dengan kepadatan yang sangat rendah.

Permasalahan yang sekarang ini adalah data yang memiliki dimensi yang tinggi. Dengan bertambahnya dimensi, data akan menjadi jarang dan mengindikasikan bahwa tiap titik akan mendekati sebuah outlier. Dengan kata lain, untuk data yang memiliki dimensi yang tinggi, perkiraan untuk menemukan outlier akan menjadi rumit.

Banyak metode yang digunakan untuk mencari outlier akan tetapi jika digabungkan dengan data yang memiliki dimensi yang tinggi, maka hanya ada beberapa metode yang dapat digunakan yaitu *Clustering-based*, *Distance-based*, dan *Density-based*.

1.2 Tujuan

Berdasarkan rumusan masalah diatas, maka tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Mempelajari metode *Clustering-based*, *Distance-based dan Density-based* dalam mendeteksi outlier.
2. Membangun perangkat lunak deteksi outlier dengan menerapkan metode *Clustering-based*, *Distance-based dan Density-based*.
3. Melakukan pengujian presentase ketepatan metode pencarian outlier pada beberapa dataset yang telah diketahui nilai kebenaran akan data anomalinya maupun yang tidak diketahui secara benar data yang termasuk data anomali. Dataset yang akan diujikan mempunyai dimensi baik rendah maupun tinggi.

2. DASAR TEORI

2.1 Deteksi Outlier

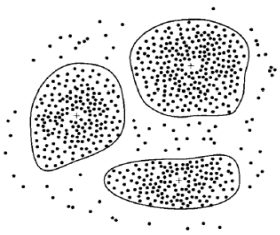
Sebuah sumber data atau dataset pada umumnya mempunyai nilai-nilai pada setiap obyek yang tidak terlalu berbeda jauh dengan obyek lain. Akan tetapi terkadang pada data tersebut juga ditemukan obyek-

obyek yang mempunyai nilai atau sifat atau karakteristik yang berbeda dibandingkan dengan obyek pada umumnya.

Deteksi oulier adalah suatu teknik untuk mencari obyek dimana obyek tersebut mempunyai perilaku yang berbeda dibandingkan obyek-obyek pada umumnya. Teknik data mining dapat digunakan untuk mendeteksi adanya suatu outlier pada sebuah dataset. Teknik data mining yang diganakan adalah Clustering-based, Distance-based dan Density-based.

2.1.1 Metode Clustering-based

Clustering merupakan salah satu teknik analisis dalam Data Mining dimana clustering melakukan pengelompokan data berdasarkan kesamaan karakteristik data. Dengan kesamaan karakteristik pada sebuah kelompok ini dapat diambil suatu informasi yang mempunyai arti dan berguna.



Gambar 1 Ilustrasi *clustering*

2.1.1.1 Algoritma CLAD

Pada CLAD terdapat dua fase utama yaitu pembuatan cluster dan meng-*assign* obyek – obyek data pada data set. Secara sederhana dapat dideskripsikan sebagai berikut:

1) inisialisasi cluster_outier = 0
2) //fase_1
3) untuk setiap cluster_outlier hitung jarak centroid cluster dengan setiap obyek data
4) jika jarak obyek data dengan centroid cluster kurang dari lebar_cluster masukkan obyek ke dalam cluster
5) jika jarak obyek data lebih dengan centroid lebih dari lebar_cluster dan obyek data belum menjadi anggota cluster_outlier lain maka buat cluster_outlier baru dengan obyek data sebagai centroid
6) //fase_2
7) untuk setiap cluster_outlier hitung jarak centroid cluster dengan setiap obyek data
8) jika jarak centroid cluster_outlier dengan obyek data kurang dari lebar cluster dan obyek data belum menjadi anggota cluster_outlier maka masukkan obyek data ke dalam cluster_outlier

2.1.1.2 Lebar *cluster*

Lebar *cluster* dideskripsikan sebagai jangkauan antara centroid cluster_outlier dengan obyek data. Perhitungan parameter lebar cluster_dilakukan dengan mengambil sampel data dari data set kemudian dihitung jarak rata-rata.

2.1.1.3 Fungsi Jarak

Perhitungan jarak antara dua obyek data dilakukan dengan menggunakan fungsi Euclidan dimana fungsi ini dapat digunakan pada dimensi yang tinggi.

$$\text{distance}(Z_1, Z_2) = \sqrt{\sum_{i=1}^{i=|Z_1|} [Z_{1i} - Z_{2i}]^2} \qquad (1)$$

2.1.1.4 Analisis *cluster*

Penentuan bahwa suatu cluster merupakan cluster outlier, CLAD menggunakan 2 attribut pada cluster yang telah terbentuk yaitu *distance* dan *density* dari cluster lain. Dikarenakan setiap cluster memiliki lebar cluster yang tetap maka kepadatan (*density*) dari setiap cluster dihitung berdasarkan jumlah obyek yang termasuk dalam cluster tersebut. Jarak (*distance*) antar cluster dihitung dengan menggunakan *average inter-cluster distance* (ICD).

$$\text{ICD}_i = \left[\sum_{j=1, j\neq i}^{j=|C|} \text{distance}(c_i, c_j) \right] \div (|C| - 1) \qquad (2)$$

Standar deviasi yang digunakan adalah *median absolute deviation* (MAD) dikarenakan persebaran jumlah anggota cluster yang tidak merata.

$$\text{MAD}(P) = median(\{|p - median(P)| : p \in P\}) \qquad (3)$$

Dengan menggunakan fungsi ICD dan MAD dapat diketahui apakah suatu cluster dikatakan sebagai cluster outlier. Cluster dengan label *sparse* dikatakan sebagai *local* outlier, sedangkan cluster dengan label *distant* dikatakan sebagai *global* outlier. Sebuah cluster dikatakan sebagai cluster outlier apabila cluster tersebut *distant* dan *sparse* yang merupakan gabungan dari *local* outlier dan *global* outlier.

$$C_{\text{distant}} = \{c_i \in C \mid ICD_i > \text{AVG}(ICD) + \text{SD}(ICD)\} \qquad (4)$$

$$C_{\text{sparse}} = \{c_i \in C \mid Count_i > \text{AVG}(Count) - \text{MAD}(Count)\} \qquad (5)$$

$$C_{\text{demse}} = \{c_i \in C \mid Count_i > \text{AVG}(Count) + \text{MAD}(Count)\} \qquad (6)$$

Sebuah cluster diakatakan sebagai cluster_outlier jika memiliki status *distant* dan *sparse.*

2.1.2 Metode *Distance-based*

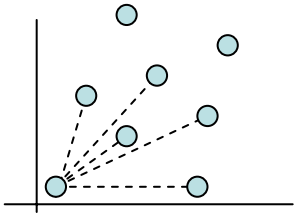
Sebuah metode pencarian outlier yang populer dengan menghitung jarak pada obyek tetangga terdekat (*nearest neighbor*). Dalam pendekatan ini, satu obyek melihat obyek-obyek *local neighborhood* yang dedefinisikan dengan *k-nearest neighbor*. Jika ketertetanggaan antar obyek relatif dekat maka dikatakan obyek tersebut normal, akan tetapi jika ketertetanggaan antar obyek relatif sangat jauh maka dikatakan obyek tersebut tidak normal.

2.1.2.1 Algoritma Bay's

Algoritma Bay's mencari outlier dengan menghitung jarak antar obyek data pada dataset. Pencarian ini dilakukan dengan membandingkan jarak yang telah dihitung dengan jarak pada *k* tetangga terdekat (*k-nearest neighbor*), kemudian dipilih untuk menjadi tetangga terdekat menggantikan tetangga terdekat yang terjauh.

2.1.2.2 Analisis obyek data

Obyek data dikatakan sebagai outlier apabila obyek tersebut memiliki obyek tetangga yang sangat sedikit pada jarak tertentu dan memiliki jarak yang jauh dibandingkan dengan jarak rata-rata obyek-obyek data tetangga terdekat.



Gambar 2 Analisa obyek data pada metode Distance-based

2.1.3 Metode Distance-based

Metode density-based tidak secara eksplisit mengklasifikasikan sebuah obyek adalah outlier atau bukan, akan tetapi lebih kepada pemberian nilai kepada obyek sebagai derajat kekuatan obyek tersebut dapat dikategorikan sebagai outlier. Ukuran derajat kekuatan ini adalah *local outlier factor* (LOF). Pendekatan untuk pencarian outlier ini hanya membutuhkan satu parameter yaitu *MinPts*, dimana *MinPts* adalah jumlah tetangga terdekat yang digunakan untuk mendefinisikan *local neighborhood* suatu obyek. *MinPts* diasumsikan sebagai jangkauan dari nilai MinPtsLB dan MinPtsUB. Nilai MinPtsLB dan MinPtsUB disarankan bernilai 10 dan 20. Akhirnya semua obyek dalam dataset dihitung nilai LOFnya.

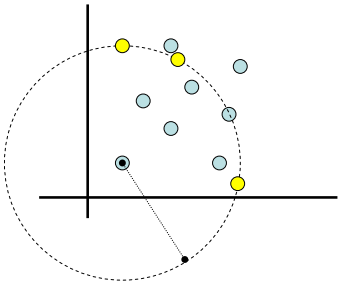
2.1.3.1 Algoritma LOF (Local Outlier Factor)

Secara sederhana algoritma LOF dapat dideskripsikan sebagai berikut:

1) menghitung jumlah tetangga terdekat
2) menghitung kepadatan lokal dari setiap obyek
3) menghitung LOF untuk setiap obyek data
4) me-*maintain* obyek-obyek data dengan nilai LOF yang tinggi

2.1.3.2 Analisis obyek data

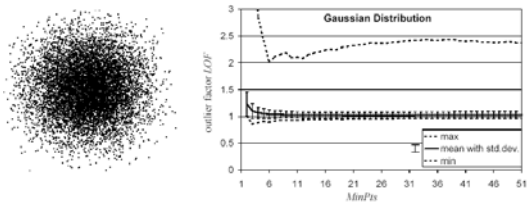
Obyek data akan dianggap memiliki nilai outlier yang tinggi jika pada jarak k tetangga terdekat memiliki kepadatan yang sangat kecil. Semakin banyak obyek – obyek tetangga dalam jarak k-tetangga terdekat, obyek ini memiliki nilai LOF mendekati 1 dan tidak seharusnya diberi label sebagai outlier.



Gambar 3 Analisa obyek data pada metode *Density-based*

2.1.3.3 Pengaruh nilai parameter *MinPts*

Algoritma *density-based* hanya membutuhkan satu parameter yaitu *MinPts*, jumlah tetanga terdekat untuk menghitung ketertetanggaan lokal



Gambar 4 Pengaruh MinPts

Gambar diatas menunjukkan obyek – obyek data didistribusikan dengan menggunakan distribusi Gaussian. Untuk setiap nilai MinPts berkisar antara 2 sampai 50, minimum, maksimum dan rata – rata nilai LOF. Karena nilai MinPts dapat berubah secara fluktuatif, maka digunakan jangkauan dari MinPts yaitu *MinPtsLB* dan *MinPtsUB* untuk mendefinisikan jangkauan terendah dan jangkauan tertinggi dari MinPts. Dengan melihat gambar 2.5 standar deviasi dari LOF hanya stabil saat MinPts mulai dari nilai 10 sampai nilai kurang dari 30.

2.2 Data

Sebuah dataset merupakan sekumpulan dari obyek-obyek data. Sebuah dataset terdiri dari beberapa dimensi data. Masing-masing dimensi data mempunyai tipe data yang berbeda antara satu dimensi dengan yang lain.

2.2.1 Jumlah data

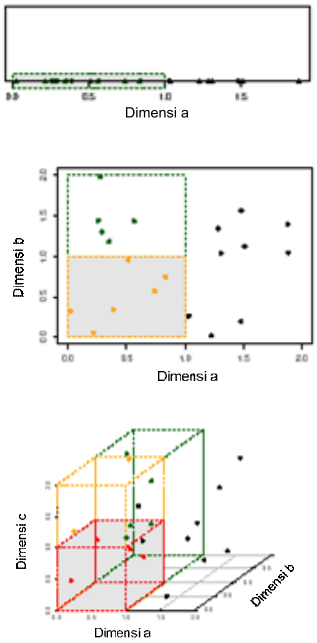
Dalam data mining permasalahan yang sering muncul adalah banyaknya jumlah data yang harus diproses untuk menemukan informasi. Peningkatan jumlah data akan berpengaruh terhadap sumber daya dan waktu untuk melakukan pemrosesan.

2.2.2 Dimensi data

Suatu dataset dapat memiliki satu atau lebih attribut atau dimensi, suatu dataset dikatakan berdimensi tinggi jika data set tersebut memiliki attribut

 ISSN: 1907-5022

yang banyak (minimal 4).



Gambar 5 Pengaruh penambahan jumlah dimensi

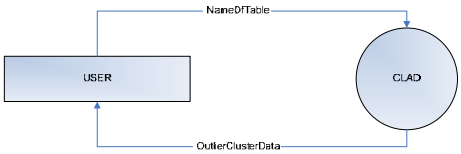
2.2.3 Tipe data

Terdapat beberapa tipe data pada dimensi sebuah data. Tipe data ini menentukan bagaimana harus memperlakukan data pada suatu operasi data.

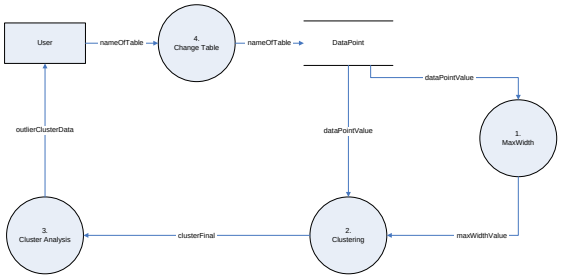
3. ANALISIS DAN PERANCANGAN

3.1 Clustering-based

3.1.1 DAD level 0

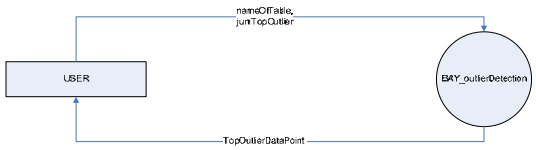


3.1.2 DAD level 1

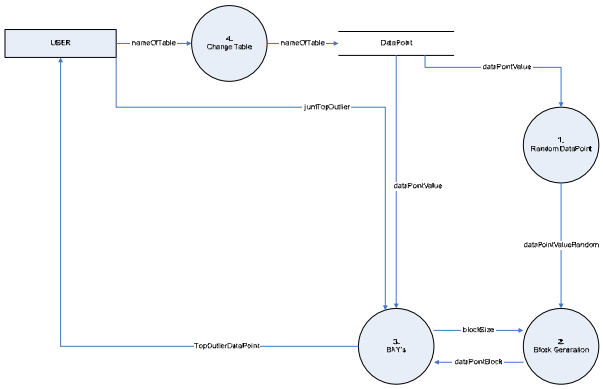


3.2 Distance-based

3.2.1 DAD level 0

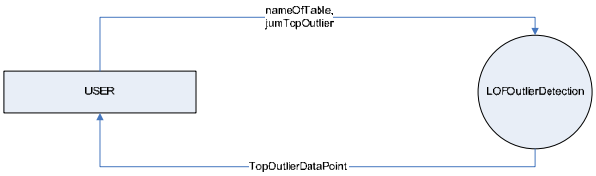


3.2.2 DAD level 1

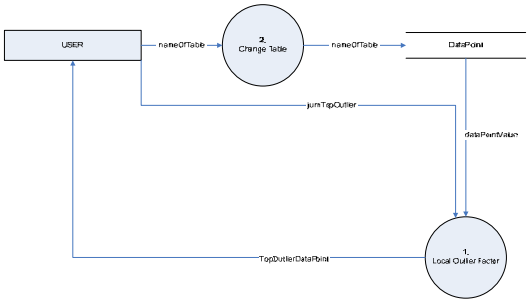


3.3 Density-based

3.3.1 DAD level 0



3.3.2 DAD level 1



4. PENGUJIAN

Pengujian dilakukan untuk melihat apakah sistem yang dibuat sudah memenuhi tujuan yang diharapkan atau belum. Pengujian ini dilakukan dengan menggunakan data sintetis, data-data riil nilai mahasiswa STT TELKOM, data bayi, dan data riil penggunaan telepon pada PT. TELKOM DIVRE II Datel Bogor.

4.1 Data Set Yang Digunakan

Dalam pengujian ini digunakan 2 data set yang bertipe numerik yaitu:

1. Data data_nilaiMahasiswa_uts_uas
2. Data data_nilaiMahasiswa_dataMining_uts_uas
3. Data data_nilaiMahasiswa_dataMining_full
4. Data data_sintetis_persebaran_1
5. Data data_sintetis_persebaran_2
6. Data data_bayi
7. Data data_telpon

Sebagian data diatas terlebih dahulu diolah sehingga selurunya bertipe numerik dan dapat digunakan dalam sistem.

4.2 Skenario Pengujian

Didalam melakukan pengujian ini mempunyai tujuan yaitu berapa baik tingkat ketepatan sistem dalam memprediksi suatu data *outlier* dan bukan *outlier*. Pengujian ini dilakukan dengan mengubah distribusi data sehingga diharapkan didapatkan presentase ketepatan sistem dalam mendeteksi *outlier*.

Adapun skenario dari pengujian terhadap dataset dapat dijelaskan sebagai berikut :

a. Pada pengujian untuk mengetahui keakuratan perangkat lunak dalam mendeteksi *outlier*, akan digunakan data set *sintetis* dan non *sintetis* yang telah diketahui obyek *outlier*-nya. Pengujian ini bertujuan untuk mengetaui performansi perangkat lunak terhadap model persebaran obyek-obyek data dalam data set.
b. Pada pengujian untuk mengetahui pengaruh bertambahnya jumlah data terhadap waktu proses deteksi, akan digunakan data set dengan jumlah dimensi yang sama dengan peningkatan jumlah data. Pengujian ini bertujuan untuk mengetahui skalabilitas perangkat lunak terhadap bertambahnya ukuran data set.
c. Untuk pengujian pengaruh dimensi data set, akan digunakan data set berukuran sama dengan peningkatan dimensi dari 1, 2, 3, sampai dengan 11.

4.3 Hasil Pengujian (Terlampir)

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

5.1.1 Sehubungan dengan hasil penelitian

1. Metode Clustering-based menunjukkan waktu deteksi yang sangat cepat akan tetapi memiliki akurasi yang kurang tinggi.
2. Metode Density-based memiliki akurasi yang lebih tinggi dibandingkan dengan kedua metode lainnya.
3. Metode Distance-based memiliki kekurangan dalam waktu proses deteksi dan hasil deteksi yang kurang akurat dibandingkan dengan metode Density-based.
4. Pada pengujian berbagai distribusi obyek data dapat disimpulkan bahwa metode *Density-based* secara umum dapat bekerja dengan baik dalam mencari *outlier*.

5.1.2 Sehubungan dengan metode Clustering-based, Distance-based dan Density-based

1. Metode *Clustering-based* akan mengembalikan klaster – klaster yang dianggap sebagai *outlier* dengan memiliki dua status pada klasternya yaitu *distant* dan *sparse*.
2. Berbeda dengan metode *Clustering-based*, metode *Distance-based* dan *Density-based* memberi nilai derajat ke-*outlier*-an pada setiap obyek data.

5.2 Saran

1. Pada metode *Clustering-based* ini salah satu hal yang sulit ditentukan adalah obyek *outlier* yang dihasilkan dalam suatu klaster.
2. Data preprosesing dalam deteksi *outlier* merupakan hal yang penting untuk diperhatikan karena data yang akan dihasilkan dalam deteksi *outlier* ini khususnya pada kasus data berdimensi sangat tinggi. Reduksi dimensi merupakan satu hal yang sangat menarik untuk diteliti lebih lanjut.
3. Masing – masing metode mempunyai kelebihan dan kekurangan dikarenakan dari perbedaan sudut pandang dalam mendeteksi obyek *outlier*. Dengan menggabungkan beberapa metode diharapakan saling menutupi kekurangan metode dengan kelebihan metode yang lain.

6. DAFTAR PUSTAKA

Aggarwal, C., Yu, P. S., Park, 2001, *Outlier detection for high dimensional data*, SIGMOD, 2001.

Edwin M. Knorr, Raymond T. Ng, *Algorithms for mining distance-based outliers in large datasets*, In Proc. 24th Int. Conf. Very Large Data Bases, VLDB, 1998.

H. A. Muhammad, K. C. Philip, 2003, *Identifying Outliers via Clustering for Anomaly Detection*, Department of Computer Sciences Florida Institute of Technology Melbourne.

Han, J., Kamber, M., 2001, *Data Mining: Concepts and Techniques*, USA: Morgan Kaufmann, Academic Press.

Hard. David, Mannila. Heikkei, Smyth. Padhraic, 2001, *Principles of Data Mining*, England: MIT Press, Cambridge Massachusetts, London.

Hawryszkiewycz. I. T, 1994, *Introduction to Systems Analysis and Design*, Australia: Prentice Hall.

Lozano. Elio, Acuña. Edgar, *Parallel Algorithms for distance-based and density-based outliers*, University of Puerto Rico Mathematics Department.

Markus M. Breunig, Hans-Peter Kriegel, Raymod T. Ng, Jorg Sander, *LOF : Identifying Density-Based Local outliers*. In ACM SIGMOD Internationa Conference on Management of Data, 2000

Otey, Matthew Eric, Parthasarathy. Srinivasan, Ghoting. Amol, *An Empirical Comparison of Outlier Detection Algorithms*, Department of Computer Science and Engineering The Ohio State University

Otey, Matthew Eric, Parthasarathy. Srinivasan, Ghoting. Amol, *LOADED: Link-based Outlier and Anomaly Detection in Evolving Data Sets*, Department of Computer Science and Engineering The Ohio State University

Papadimitriou. Spiros, Kitagawa. Hiroyuki, Gibbons. Phillip B., Faloutsos. Christos, *LOCI: Fast Outlier Detection Using the Local Correlation Integral*.

Stephen D. Bay, Mark Schwabacher, *Mining Distance-Based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule*. In ACM SIGMOD 2003

Sudipto Guha, Rajeev Rastogi, Kyuseok Shim, *CURE: An efficient clustering algorithm for large databases*. In ACM SIGMOD Internation Conference on Management of Data, 1998.

Tan. Pang-Ning, Steincbach. Michael, Kumar. Vipin, 2006, *Introduction to Data Mining*, Pearson Education Inc.

LAMPIRAN
Hasil Pengujian
A. Akurasi

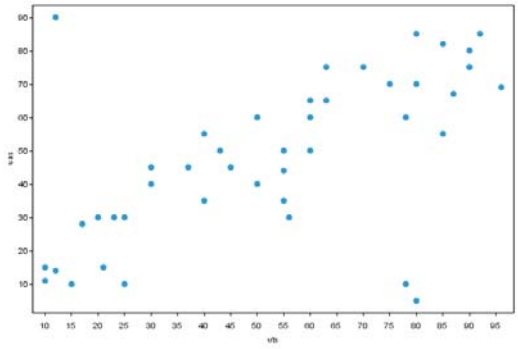
Pengujian dilakukan sebanyak 6 kali pengujian dengan menggunakan data set:

1. Data data_nilaiMahasiswa_uts_uas
2. Data data_nilaiMahasiswa_dataMining_uts_uas
3. Data data_nilaiMahasiswa_dataMining_full
4. Data data_sintetis_persebaran_1
5. Data data_sintetis_persebaran_2
6. Data data_bayi

Pada pengujian ini menggunakan data set berdimensi rendah yang diharapkan memiliki kemiripan akurasi pada data set berdimensi tinggi. Data set pengujian memiliki model distribusi yang berbeda diantara masing-masing data set.

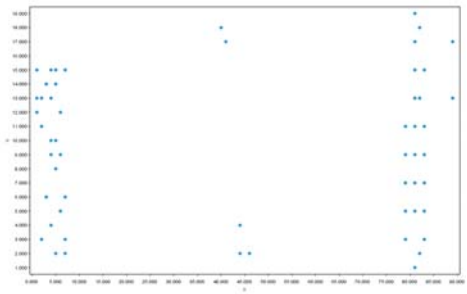
Berikut adalah contoh data set:

Data set niliai mahasiswa data set ini mendeskripsikan nilai UTS dan UAS mahasiswa. Data set ini terdiri atas 2 dimensi yaitu nilai UTS dan UAS serta memiliki 44 baris data.



Gambar 6 Model Distribusi Data Nilai Mahasiswa UTS dan UAS

Data set sintetis terdiri atas 2 dimensi yaitu koordinat sumbu X dan Y serta memiliki 55 baris data.



Gambar 7 Model Distribusi Data sintetis

Hasil pengujian yang diperoleh setelah melakukan pengujian pada semua data set yang tersedia adalah bahwa secara umum pada pemodelan berbagai distribusi data secara umum, metode *Density-based* memiliki akurasi yang lebih tinggi dibandingkan dengan metode *Clustering-based* dan *Distance-based*.

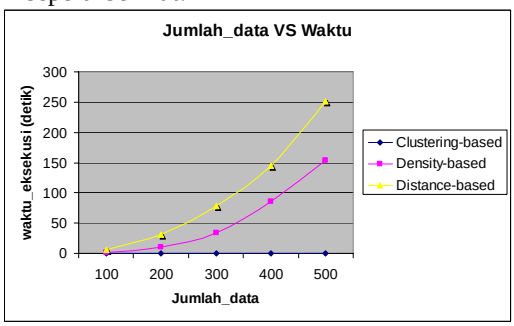
B. Pengaruh Jumlah Data Terhadap Waktu Eksekusi

Pada pengujian yang bertujuan untuk mengetahui pengaruh jumlah data terhadap proses eksekusi dalam mencari *outlier* digunakan dataset dengan jumlah dimensi tetap yaitu 12 dimensi dan jumlah data 100, 200, 300, 400 dan 500.

Tabel 1 Perbandingan pengujian jumlah data terhadap waktu eksekusi

Record	Clustering	Density	Distance
100	**0.021**	**1.969**	**6.350**
200	**0.038**	**10.728**	**31.625**
300	**0.063**	**34.322**	**77.940**
400	**0.085**	**85.976**	**145.409**
500	**0.106**	**153.534**	**250.981**

Dari data tabel pengujian diatas dapat dibuat grafik seperti berikut:



Gambar 8 Grafik perbandingan peningkatan jumlah data terhadap waktu eksekusi

C. Pengaruh Jumlah Dimensi Terhadap Waktu Eksekusi

Pada pengujian yang bertujuan untuk mengetahui pengaruh jumlah data terhadap proses eksekusi dalam mencari *outlier* digunakan data set dengan jumlah data tetap yaitu 200 baris data dan jumlah dimensi yang bertambah dari 1 sampai dengan 12.

Tabel 2 Perbandingan pengujian jumlah dimensi terhadap waktu eksekusi deteksi *outlier*

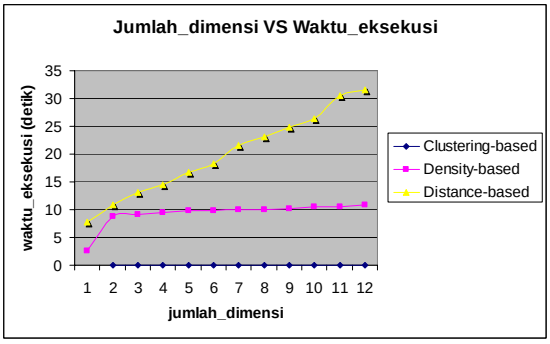
Dimensi	clustering	density	Distance
1	**-**	**2.625**	**7.741**
2	**0.053**	**8.816**	**10.819**
3	**0.069**	**9.178**	**13.131**
4	**0.078**	**9.484**	**14.531**
5	**0.078**	**9.825**	**16.719**
6	**0.035**	**9.813**	**18.247**

ISSN: 1907-5022

Tabel 2 Perbandingan pengujian jumlah dimensi terhadap waktu eksekusi deteksi *outlier* (lanjutan)

7	**0.031**	**9.997**	**21.619**
8	**0.031**	**10.062**	**23.182**
9	**0.041**	**10.231**	**24.865**
10	**0.035**	**10.465**	**26.359**
11	**0.034**	**10.597**	**30.541**
12	**0.041**	**10.809**	**31.628**

Dari data tabel pengujian diatas dapat dibuat grafik seperti berikut:



Gambar 9 Grafik perbandingan peningkatan jumlah dimensi terhadap waktu eksekusi deteksi *outlier*