Pengelompokan Selari Untuk Data Skala Besar dan Dimensional Tinggi Pada Aplikasi Perlombongan Data

Refcan Afivi, PM.Dr.Mohd Noor Md Sap
Telp:+628126776368, +62751202695
refcan@semenpadang.co.id

Faculty of Computer Science and Information System, Universiti Teknologi Malaysia
81310 UTM Skudai, Johor, Bahru, Malaysia

Abstrak *: Sejumlah data besar untuk penemuan pengetahuan adalah perlombongan informasi, yang mempunyai perkakas canggih untuk membongkar aturan asosiasi, pola kecenderungan, mendeteksi penyimpangan, pengelompokan, penggolongan informasi, dan perkembangan kaedah bersifat prediksi. Dengan peningkatan ukuran datadan berdimensi tinggi maka pengolahan teknik secara selari perlu diberlakukan bagi menguasaan perkakas dalam penemuan pengetahuan untuk menglasilkan informasi bermanfaat dari set data berskala besar dan mengurangi waktunya untuk analisa. Pada kesempatan ini akan diperkenalkan suatu kerangka grid mudah suai yang tidak hanya mengurangi perhitungan, grid pengelompokan berdasar kepadatan data untuk meningkatkan mutu pengelompokan.Density-parallel merupakan algoritma pengelompokan yang tidak diawasi, karena algoritma inihanya tergantung pada* α *dan* β *ditandai dengan besarnya penyimpangan nilai histogram yang tersebar purata. Pada Density-paralel dapat dilakukan secara selari data dan selari tugas yang bermanfaat untukl mempercepat proses dengan laju tinngi serta mengurangi biaya komunikasi dalam implementasi multi pemproses secara selari.*

Keywords: : Data Mining, Parallel, Clustering, CLIQUE, High Dimensional Data,

1. Pengenalan

Penggunaan kaedah perlombongan data untuk menghasilkan maklumat yang bermanfaat kini semakin diminati oleh pengambil keputusan untuk mengkomersialkan hasil pengeluarannya, kebolehan dari kaedah ini untuk menggali corak-corak dan ilmu yang tersembunyi didalam data telah menyakinkan pengguna untuk mengambilnya sebagai alat untuk penganalisaan data yang lebih tepat [06,13,18,22,40,44], proses ini merupakan suatu penyaringan maklumat bermanfaat dari data yang tidak dikenal pada mulanya, seperti maklumat ilmu pengetahuan, peraturan, pengkelasan dan pengelompokan, penyaringan ini tentunya dilakukan dengan teknik yang terpola sehingga data tersebut dapat diproses secara tepat [18,22,40].

Secara umum perlombongan data merupakan suatu bagian proses dari *Knowledge Discovery Data* [KDD] atau proses penemuan pengetahuan dalam pangkalan data dimana kedua-duanya digambarkan bukan suatu proses sederhana dalam pengenalan dan berpola teladan yang bermanfaat [19,28,29,30], didalam proses penemuan pengetahuan pada pangkalan data terdapat langkah-langkah yang meliputi : (i) Pembersihan data [37,40], (ii) Pengintegasian data [37,40], (iii) Pemilihan data [37,40], (iv) Perubahan bentuk data [37,40], (v) Perlombongan data [37,40], (vi) Penilaian pola [37,40], (vii) Penyajian pengetahuan [37,40].

Sampai saat ini para pengkaji telah banyak mengeluarkan ide-idenya untuk mengatasi permasalahan perlombongan data terhadap pangkalan data yang sangat besar, namun masih perlu dilakukan penyempurnaan-penyempurnaan khususnya dalam pengelompokan secara otomasi yang mengubah bentuk data kedalam pengetahuan dan maklumat dari sejumlah tumpukan data dengan memanfaatkan keunggulan teknik pengelompokan secara selari dengan tujuan meningkatkan laju proses dan ketelitian tinggi.

1.2 Latar belakang permasalahan

Untuk mempercepat laju proses dan meningkatkan efektifitas terhadap pengelompokan data terhadap data yang berjumlah sangat besar hanya dapat diatasi dengan menggunakan teknik pintar, teknik perselarian dan penyederhanaan pada algoritma yang digunakan, dengan demikian untuk melakukan kajian terhadap pengelompokan data, masih sangat diperlukan agar tumpukan data tersebut dapat diproses dan dimanfaatkan secara utuh dengan mempertimbangkan waktu dan kualiatas yang dihasikan, disamping itu sejumlah teknik pengelompokan yang ada masih mempunyai beberapa kelemahan yang dirasakan [21,29]

Kelemahan yang paling dirasakan dalam pengelompokan data yang berjumlah sangat besar seperti, (i) Memerlukan waktu yang cukup lama dalam memproses data menjadi maklumat yang berguna [21,29], (ii) Untuk memproses data menjadi maklumat yang berguna masih menggunakan kaedah secara linear sehingga kemampuannya terbatas untuk

pengelompokan data berukuran sangat besar [21,28] , (iii) Belum menggunakan perlombongan data secara pintar dan pengelompokan secara selari[19,32], sesuai dengan pendapat J.Han and M.Kember[06] bahawa permasalahan terbesar dalam perlombongan data adalah proses perselarian dan pengagihannya.

1.3 Pernyataan masalah

Dalam rangka menyelesaikan masalah tersebut diatas pada kajian ini diusulkan suatu teknik untuk pengelompokan data pada tumpukan data yang berjumlah sangat besar yang dapat diproses secara berulang-ulang dengan laju tinggi:

1. Kaedah apa yang sesuai untuk pengelompokan data agar dapat digunakan pada tumpukan data sangat besar dengan berlaju tinggi.
2. Bagaimana meningkatkan laju dalam pengelompokan data terhadap tumpukan data yang berjumlah sangat besar dengan perselarian yang bekerja secara automasi tanpa masukan parameter input dari pengguna.
3. Bagaimana agar pengelompokan Selari DENSITY-PARALLEL yang diusulkan dapat meningkatkan laju proses terhadap tumpukan data yang berjumlah sangat besar, dan apa peralatan yang mendukung kaedah tersebut

1.4 Objek Kajian

Sasaran utama dari kajian ini adalah (i) Menyelidiki perlombongan data terhadap kaedah pengelompokan data berjumlah sangat besar, (ii) Mengusulkan suatu algoritma pengelompokan data menggunakan perselarian yang tidak memerlukan input parameter dari pengguna, (iii) menyempunakan algoritma CLIQUE agar dapat memproses pengelompokan dengan laju dan keterilian yang tinggi, sasaran dari kajian ini adalah sebagai berikut:

1. Untuk menyelidiki langkah dan proses pengelompokan secara selari dalam rangka meningkatkan kualiti dan laju proses untuk mendapatkan maklumat baru dari tumpukan data yang sangat besar.
2. Membuat suatu kaedah algoritma pengelompokan selari DENSITY-PARALLEL dalam rangka mempercepat laju.
3. Menyempurnakan Alogritma CLIQUE untuk diterapkan pada DENSITY-PARALLEL guna terhadap meningkatkan kualiti pada proses pengelompokan data terhadap data yang berjumlah besar.

1.5 Skop Kajian

Dalam melakukan kajian ini perlu dibuat skop lebih jelas dan terfokus agar sasaran yang diharapkan dalam menciptakan kaedah baru terhadap perlombongan data yang bekerja secara selari dapat lebih sempurna. Adapun skop kajian ini adalah sebagai berikut:

1. Studi Perbandingan dan peninjauan ulang terhadap pengelompokan data dalam rangka menemukan kelemahan dan kelebihan kaedah tersebut, iaitu : (i) perlombongan data, (ii) proses perlombongan data, (ii) kaedah pengelompokan data: pengelompokan heirarki:Aglomeratif, CHAMELEON, pengelompokan partisi:CLARANS, BIRCH, K-MEANS, pengelompokan kategorikal:ROCK, CACTUS, pengelom pokan berdasarkan ketumpatan dan grid:DBSCAN, DBCLASD,OPTIC,CLIQUE, pengelompokan selari:K-MEANS, (iii) proses perselarian: selari:SPMD, selari MPMD
2. Merancang kaedah baru iaitu pengelompokan selari *unsupervisi* terhadap perlombongan data dalam rangka meningkatkan dan penyempurnaan dari kaedah sebelumnya.
3. Penaksiran pencapaian hasil dari kaedah yang diusulkan dalam rangka mengetahui kelemahan dan kelebihan.

1.6 Sumbangan Kajian

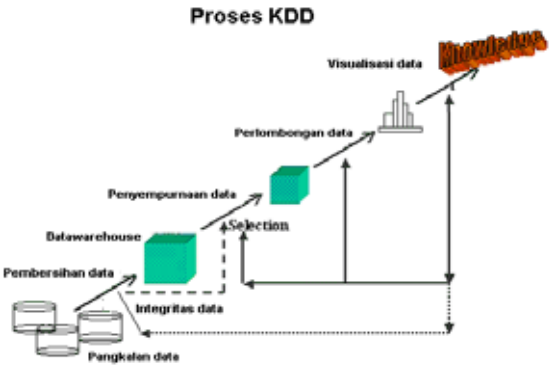
1. Suatu penyempurnaan kaedah baru dalam pengelompokan data selari yang terotomasi dengan harapan dapat memberikan suatu pemecahan untuk memproses tumpukan data yang berjumlah sangat besar.
2. Dapat memperkaya kaedah dalam perlombongan data yang ada

2.1 Perlombongan Data

Dalam perkembangannya perlombongan sejak data tahun 1990 telah mengalami kemajuan yang cepat seiring dengan keperluan pengguna, dengan kemajuan teknologi sekarang perlombongan data menjadi peranan penting dalam dunia perbisnisan mencakup berbagai aspek penggunaan seperti keperluan pembayangan, perangkaan, sain, pengkajian mesin teknologi pangkalan data dan kegunaan lainnya, perlombongan data merupakan salah satu langkah dalam proses KDD yang menentukan dalam penyaringan maklumat dari data mentah sampai maklumat berguna, dalam banyak hal perlombongan data dapat dikelompokan berdasarkan fungsinya iaitu (i) Penemuan ilmu pengetahuan (KDD)[16] : fungsi ini bertugas untuk menemukan ilmu dan pengetahuan yang bermanfaat dengan pola-pola tertentu serta pembayangannya kepada pengguna, (ii). Peramalan [13]: fungsi ini bertugas untuk menyaring pola-pola yang ada untuk peramalan kejadian dimasa akan datang, (iii). *Analisa forensic* [06,09]: fungsi ini bertugas untuk menyaring pola-pola atau menemukan pola-pola yang tidak biasa dan keganjilan didalam data.

Pada rajah 2.1 terdapat beberapa proses untuk merobah data mentah menjadi maklumat yang bermanfaat iaitu dengan melakukan tiga tahap pra-proses: (i) Pembersihan data [06,19,37]: penyimpanan data didalam pangkalan data berkemungkinan berisi data-data yang tidak berhubungan, tidak beraturan, hilang , tidak lengkap atau kemungkinan data rusak sebelum diproses ketingkat selanjutnya, tahap ini harus dilakukan agar pola data yang dibentuk dari data dapat memberikan maklumat yang benar, (ii) Penyatuan data [11,18,19,22,37] hasil dari pembersihan data akan lebih akurat apabila tingkat berbagai data tersebut menyatu kedalam satu sumber, (iii) Pimilihan data [11,18,37]: hasil dari penyatuan data kemudian diseleksi dan dianalisa untuk tentukan data berkaitan dan data tidak berkaitan agar data dapat diproses pada tingkat selanjutnya, (iv) Penukaran/penyempurnaan data [11,28.27] : Data yang terseleksi kemudian ditukar kedalam corak yang cocok pada perlombongan data dengan menunjukan ringkasan atau fungsi yang tepat, (v) Selanjutnya tahap proses iaitu tahap

perlombongan data [28,30,34], Aturan asosiasi, klasifikasi dan pengelompokan, (vi) Kemudian tahap pos proses bertugas untuk penafsiran hasil yang telah diperolehi iaitu : Analisis [27,28]: untuk menganalisa pola-pola sesuai yang merupakan hasil dari tahap proses, Pembayangan [11,27,28]: untuk menampilkan hasil yang telah diperoleh pada tahap awal sampai tahap terakhir



Rajah 2.1 Proses KDD dalam Pangkalan data

2.2 Pengelompokan Data

Pengelompokan data akan menjadi sukar apabila melakukan pengolahan data yang sangat besar dan banyak atribut dari jenis yang berbeza, sehingga sering terjadi pemaksaan komputasi pada algoritma pengelompokan yang ada. Berbagai algoritma baru-baru ini muncul memperlihat-kan dan dengan sukses dilakukan bagi pengelompokan data. Kajian ulang tentang pengelompokan data diuraikan sebagai berikut.

2.2.1 Pengelompokan Hirarki

M.Halkidi, [09] Mengemukankan pengelompokan hierarki adalah pembentukan pengelompokan secara hirarki dengan kata lain disebut juga pengelompokan pohon dikenal dengan dendrogram, tiap-tiap tangkai pohon pengelompokan berisi pengelompokan anak, pengelompokan menyiapkan point-point yang diperlu oleh induk mereka, pendekatan seperti itu pengizinkan penyelidi data tentang tingkatan kegranulan yang berbeza. Kaedah pengelompokan hierarki digolongkan kedalam *agglomerative* (data atas kebawah) dan bersifat memecah belah, [06,201]. Suatu aglomerative mengelompokan mulai dengan satu hal secara berulang menggabungkan dua atau lebih pengelompokan paling sesuai, pengelompokan bersifat memecah belah mulai dengan pengelompokan semua data yang ditunjuk secara berulang mencari pengelompokan paling sesuai untuk dilanjutkan sampai kesuatu tempat terhenti. Biasanya ada dua pendekatan yang digunakan dalam algoritma ini yaitu pendekatan dari bawah ke atas: pengelompokan terdiri dari n 1 elemen pengelompokan memberikan n objek dan tidak lebih dari pengelompokan kasar dimana satu pengelompokan terdiri dari semua n objek dengan algoritma seperti berikut:

INPUT . satu populasi o,
. fungsi satuan jarak $D := P(o) \times P(o) \rightarrow [\, d_{min}, d_{max}]$

langkah 1 Mulai dengan tempat pengelompokan
$C^o := \{\{\chi\}|\chi \in 0$

Rajah 2.7 :Algoritma heirarki (pendekatan bawah ke atas)

Keuntungan dari pengelompokan hirarki meliputi: fleksibilitas yang ditemplekan pada tingkatan dalam penanganan segala format jarak atau bersamaan kesesuaian pada atribut, sedangkan kerugiannya dihubungkan dengan ketidak jelasan ukuran-ukuran pemberhentian, faktanya menunjukan bahawa algoritma hirarki tidak mengunjungi balik ketika membangun antara pengelompokan dengan tujuan peningkatan[48].

2.2.2. Pengelompokan Berdasarkan Partisi

Algoritma pengelompokan menyekat membangun suatu partisi objek-objek di dalam pangkalan data ke dalam pengelompokan objek-objek lebih serupa satu sama lain dibanding dengan objek-objek di dalam pengelompokan berbeza. Kaedah k-means dan k-medoid menentukan k wakil pengelompokan dan menugaskan masing-masing objek kepada pengelompokan dengan wakilnya yang terdekat ke suatu objek, seperti penjumlahan jarak kuadrat antara objek-objek dan wakil mereka diperkecil. Algoritma k-modes perluasan dari algoritma k-means ke daerah mutlak [A26,15,18,20,21]. CLARANS [151,A34]. dapat dianggap sebagai perluasan partisi dalam pangkalan data. Proses pengelompokan di dalam [151,A34] adalah dapat disamakan untuk mencari suatu grafik untuk menemukan suatu nilai jumlah maksimum fungsi biaya, dimana nod masing-masing adalah suatu solusi potensi (satu set k-mediods). BIRCH [A41,51,52,53], adalah suatu algoritma pengelompokan dengan penekanan ke luar dari set data inti. Pendekatan mereka memperkenalkan konsep suatu CF pepohon (*Clustering Feature* / pengelompokan corak), suatu pepohon seimbang, berapa jumlah maksimum anak-anak dapat dikendalikan oleh suatu faktor bercabang, B. Juga garis tengah maksimum sub pengelompokan pada nod masing-masing dapat dikendalikan oleh suatu parameter ambang pintu. CF pohon membangun dengan dinamis sebagai data poin-poin disisipkan memastikan bahwa suatu pohon seimbang dibentuk. Biaya-biaya I/O BIRCH algoritma telah ditunjukkan untuk menjadi $O(N)$ dan algo

ritma ini juga menghasilkan kwalitas baik untuk pengelom pokan dengan ketidakpekaan kepada order masukan poin-poin sebagai dilaporkan di dalam [A41,51, 52, 53].

2.2.3 Pengelompokan kategorikal

Pengelompokan data kategorikal sering memerlukan suatu pendekatan berbeza dibandingkan dengan pengelompokan menunjuk poin data. Perbedaan ini dalam kaitan dengan ketidak hadiran tentang segala atribut tertentu, Sebagai contoh mempertimbangkan transaksi penjualan dari suatu perusahaan rombongan mobil, masing-masing transaksi berisi informasi lengkap dari tiap transaksi penjualan seperti taip mobil, warna mobil, merek mobil, dan lainnya. Masing-masing atribut dapat mengambil beberapa nilai atribut. Taip suatu mobil yang manapun Toyota, Mercedez, Suara/audio atau merek lain dan dengan cara yang sama masing-masing mobil dijual boleh mengambil warna manapun dari di antara satuan warna tersedia. Suatu informasi menarik yang dapat digali dari data ini adalah pengelompokan pilihan pelanggan manapun taip warna, merek, dan lainnya.

Bagaimanapun, nilai-nilai atribut tidak mempunyai pesanan dikenakan pada batasnya, jarak yang didasarkan kaedah pengelompokan gagal untuk diterapkan. k-modes [A26,30] adalah salah satu dari pekerjaan awal pengelompokan kategorikal. STIRR[A19,A20,A21] adalah suatu algoritma pengelompokan katekorikal berlaku spektral yang disamaratakan teknik menyekat kepada permasalahan dalam "hyper-graph" pengelompokan. Kategorikal set data diperagakan sebagai "hyper-graph" dan suatu kaedah pengelompokan berdasar pada sistem dinamis tidak linier yang diberlakukan bagi "hyper-graph". Suatu "hyper-graph" algoritma penyekatan digunakan untuk pengelompokan kategorikal set data di dalam [A25]. ROCK[A22,A23] adalah suatu kategorikal algoritma pengelompokan set data yang menggunakan suatu konsep roman menghubung pautan antara poin-poin data. Point data diperlakukan sebagai tetangga jika persamaan mereka melebihi suatu ambang pintu dan banyaknya pautan antara dua poin data adalah banyaknya tetangga umum yang terbagi. CACTUS [A18] adalah suatu "summarization-based" algoritma pengelompokan terbaru mempunyai keperluan I/O minimum dan didasarkan pada menemukan inter-attribut dan ringkasan intra-attribut.

2.2.4 Pengelompokan Berdasarkan Ketumpatan

DBSCAN adalah suatu ketumpatan yang didasarkan algoritma pengelompokan [A15]. Kerana masing-masing objek suatu pengelompokan dalam lingkungan ditentukan dengan radius, r harus berisi sedikitnya suatu jumlah minimum poin-poin, $\in$. Di sini kedua-duanya, radius dan jumlah minimum poin-poin adalah parameter masukan yang sangat besar mempengaruhi mutu pengelompokan. Bagaimanapun, pengelompokan dari bentuk berubah-rubah dapat ditemukan di dalam pendekatan ini dan sungguh cocok untuk mengenai ruang pangkalan data. DBCLASD adalah suatu distribusi algoritma pengelompokan [A40] didasarkan pada asumsi menunjuk suatu pengelompokan seragam yang dibagi-bagikan. DBCLASD tidak memerlukan masukan pemakai, tetapi bagaimanapun mempunyai suatu efisiensi lebih rendah dibanding algoritma DBSCAN. OPTIC adalah suatu ketumpatan terbaru yang didasarkan pengelompokan algoritma [191,192,A01]. OPTIC bagaimanapun tidak menghasilkan suatu pengelompokan set data dengan tegas tetapi sebagai gantinya menciptakan suatu pemesanan ditambahkan pada pangkalan data untuk mewakili pengelompokan berbasis ketumpatan strukturnya.

2.2.5 Pengelompokan subspace

Kemampuan untuk temukan pengelompokan menempelkan subspaces data skala besar dan dimensional tinggi, serta bisa dimengerti pemakai akhir hasil. CLIQUE[211], mengidentifikasi pengelompokan padat di dalam subspaces dimensionalas secara maksimum dan menghasilkan uraian pengelompokan dalam ungkapan yang diperkecil. Algoritma CLIQUE menunjukan secara efisien temukan pengelompokan akurat didalam dimensional tinggi.

Subspace pengelompokan dapat bermanfaat bagi aplikasi lain di samping penambangan data seperti OLAP, sebagai contoh: ruang data yang pertama disekat ke dalam unit padat dan daerah jarang [211]. Data di dalam daerah padat disimpan disuatu array sedangkan suatu struktur pohon digunakan untuk daerah jarang. sekarang ini, para pemakai diperlukan untuk menetapkan unit padat dan jarang dalam ruang dimensi. Dengan cara yang sama kaedah penghitungan untuk query diatas OLAP pada kubus data memerlukan identifikasi dari daerah padat di dalam kubus data jarang. CLIQUE dapat digunakan untuk permasalahan dalam mengevaluasi mutu pengelompokan di dalam subspace berbeda. Satu pendekatan untuk memilih pengelompokan dengan memaksimalkan perbandingkan kepadatan pengelom pokan di atas ruang dimensi, CLIQUE menyelidiki pendukungan sistem kepada pemakai untuk memilih parameter model, r dan t, suatu pendekatan alternatif untuk menemukan unit padat.

Evaluasi empiris menunjukkan algoritma CLIQUE secara siri dengan ukuran masukan dan mempunyai skala baik menemukan pengelompokan yang ditempelkan untuk menurunkan dimensional subspaces, walaupun tidak ada pengelompokan di dalam ruang data yang asli. Selanjutnya kelayakan subspace pengelompokan siri haruslah dipertimbangkan terhadap suatu penglompokan secara selari dengan mengurangi parameter masukan dari pengguna.

2.2.6 Pengelompokan selari.

Algoritma pengelompokan selari dapat dibagi menjadi tiga jenis: pengelompokan hirarki, pengelompokan distance-based dan pengelompokan kepadatan[120]. Secara umum, algoritma pengelompokan mempekerjakan dua pekerjaan iaitu: pengulangan luar di atas anggota pengelompokan yang mungkin dan suatu pengulangan bagian dalam untuk kemungkinan cocok dijadikan pengelompokan terbaik dengan jumlah ditentukan. Pengelompokan distance-based,

pengelompokan dibangun dengan komputasi suatu solusi optimal untuk meminimumkan jumlah jarak di dalam pengelompokan data. Ini dilaksanakan diawali dengan membangun sejak semula solusi baru atau dengan penggunaan suatu solusi pengelompokan sah sebagai titik awal untuk peningkatan kesamaan. Kesamaan di dalam kaedah distance-based pengelompokan dapat dimanfaatkan kedua-duanya dalam tingkatan luar, dengan berusaha angka-angka pengelompokan berbeda secara selari, dan di tingkatan bagian dalam dengan komputasi ilmu tentang meter jarak didalam selari

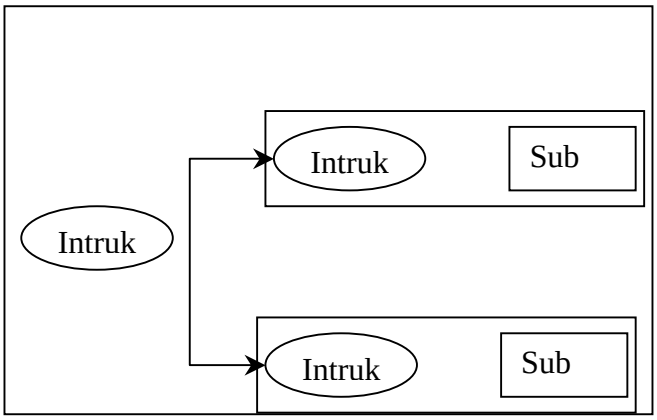
Dalam kes selari algoritma yang diusulkan DENSITY-PARALEL akan memberikan suatu selusi tentang kaedah pengelompokan selari yang praktis untuk data yang sangat besar dan merupakan pengelompokan ***unsupervisi*** dari pemakai, sehingga dapat memudahkan proses pengelompokan data.

2.3. Perselarian

Pengelompokan data adalah suatu antar displin ilmu yang mempunyai fungsi aplikasi didalam area berbeza seperti : bioinformatics, medis informatics, scientic data analisa, financial analisa, konsumen profile, dan lain-lain, pada setiap daerah aplikasi jumlah data yang tersedia untuk analisa, Tahun terakhir ditemui bahawa skalabilitas data pada pekerjaan pengelompokan merupakan suatu faktor kritis. Sampailah pada versi selari dari umumnya kaedah pengelompokan data mulai dikembangkan untuk mengatasi permasalahan terhadap penglompokan data yang berskala besar dan berdimensi tinggi yang mempunyai laju tinggi. Aplikasi selari adalah suatu tantang didalam penggunaan yang luas dari pengelompokan data, untuk itu telah di tinjauan ulang terhadap proses selari sebagai berikut:

2.3.1. Selari Data dan Selari tugas

Pertama D.Taniar& K.Smith, (2001). [46] yang mengusulkan proses selari sebagai suatu keharusan didalam lingkungan yang mempunyai pangkalan data sangat besar untuk menemukan pengetahuan menarik seperti pola teladan, asosiasi, perubahan, pengelompokan, keganjilan data dan struktur penting. Sasaran dari kaedah pengelompokan selari untuk memotifasi pengolahanan data dengan laju tinggi walaupun volumen data yang sangat besar. Seperti data-data geografis, data ilmiah, data pemeliharaan pabrik. D.Taniear & K.Smith [46] menyatakan bahwa salari mempunyai dua taip iaitu : selari data dimana adanya kesamaan data dan selari tugas mengacu pada pelaksanaan dari instruksi atau operasi serupa pada data dan waktu yang sama, semua pengolah melaksanakan program yang sama, tetapi program dapat menggunakan struktur kendali *if-then-else* untuk menentukan instruksi yang mana untuk dilaksanakan dan dilakukan dipengolahan yang mana, ada beberapa bagian menyangkut program yang sama tidak dieksekusi oleh program yang pengolah tugas lain. Selanjutnya selari control dan tugas dimana didalam kesamaan data diperlukan untuk menyekat data ke dalam subset , kemudian laju tinggi dapat dicapai dengan mengurangi jumlah data yang perlu untuk ditangani oleh masing-masing pengolah seperti terlihat pada rajah 2.12



Rajah 2.12: Selari tugas

Pada rajah diatas terlihat satu perintah diberikan pada masa pemproses (CPU), kemudian perintah tersebut diteruskan pada setiap pemproses yang ada, dimana pemproses menjalankan arahan tersebut dengan data masing yang telah di pecah-pecah sesuai dengan kapasitas pemproses, jadi data awal dipecah menjadi sub data sesuai dengan kemampuan pemproses yang ada.

Keuntungan dari teknik selari ini adalah: kemudahaan dalam implementasinya, lebih sedikit tergantung pada reka bentuk perangkat keras, lebih produktif serta terskala, cocok untuk pengolahan terhadap tumpukan data yang berjumlah sangat besar, sedangkan kelemahaannya keseimbangan beban data menjadi penentu keberhasilan laju satu pemproses.

2.3.2. Selari kontrol

Teknik ini diperlukan untuk suatu program yang baru untuk dikembangkan agar dapat mengakomodasi keperluan perselarian ini, dimana algoritma harus direka bentuk sesuai dengan reka bentuk mesin sehingga teknik ini menggunakan reka bentuk rancang spesifik untuk mencapai kesamaan, di dalam selari kontrol kesamaan sub-tasks tergantung data dan dapat mempunyai lebih beberapa data dan tidaklah untuk meningkatkan banyaknya sub arahan karena sub instruksi sudah ditetapkan, untuk menjalankan sebuah arahan diperlukan suatu algoritma secara selari, teknik perselarian ini mencoba untuk mengurangi kompleksitas tugas di dalam algoritma serial dengan pembagiannya ke dalam sub arahan sehinga dapat meningkatkan biaya koordinasi tinggi.

2.4. Peluang kajian

Di dalam bagian ini kita sudah meninjau pekerjaan yang terkait dengan pengelompokan, walaupun algoritma yang sudah ada masih mampu memproses data besar namun belum memberikan beberapa keuntungan bagi penguna, untuk itu perlu dilakukan penyempurnaan:

a. Perlu penyempunaan algoritma pengelompokan CLIQUE agar dapat bekerja secara selari, dalam rangka untuk

mencapai laju tinggi dan kemampuan yang besar dalam mengolah data berskala besar dan berdimensional tinggi.

b. Penambahan algoritma baru agar proses pengelompokan dapat bekerja secara automatis, tidak terlalu memerlukan parameter masukan, seperti: ukuran grid dan suatu ambang pintu kepadatan global.

c. Agar proses kaedah perselarian perlu dilakukan dengan pemilihan kaedah selari yang menggunakan biaya rendah dan tidak tergantung dengan keberadaan peralatan khusus.

2.5. Rangkuman

setelah melakukan kajian literatur tentang perlombongan yang difokuskan pada pengelompokan, dapat disimpulkan baha algoritma yang ada masih terbatas untuk melakukan pengelompokan terhadap data berskala besar dan dimensional tinggi, walaupun ada algoritma yang dapat memprosesnya namun algoritma tersebut masih memerlukan parameter masukan dari pengguna sehingga hal ini sangat menyulitkan dalam menentukan kebenaran parameter tersebut, untuk itu selusi yang ditawarkan iaitu dengan merobah algoritma CLIQUE agar dapat bekerja secara selari dan tanpa memerlukan parameter masukan (unsupervisi), dan algoritma tersebut tidak terikat dengan peralatan khusus, selajutnya dengan kaedah yang diusulkan agar dapat bekerja secara selari mengunakan baiaya rendah.

3.1 Merancang kajian

Perancangan kajian ini berdasar pada pendekatan bersifat percobaan dan ilmiah, yang meliput dua sub bidang yakni algoritma dan perselarian. Didalam algoritma yang diusulkan mengelompokan data selari dilakukan dengan cara selari data dan selari proses. Proses perselarian digunakan dalam lingkungan jejaringan komputer yang saling berhubungan.

Kajian lebih mendalam iaitu merupakan kembangan algoritma CLIQUE, dimana semua algoritma ini dihadapkan pada persolan untuk memproses pengelompokan data yang berskala besar dan berdimensi tinggi, disamping itu proses algoritma ini masih dilakukan secara siri sehingga memerlukan resource yang sangat besar agar algoritma ini dapat dipergunakan secara luas, secara tersusun kerangka proses dapat dilihat sebagai berikut:

Pertama: untuk melakukan pengkajian mendalam terhadap algoritma “CLIQUE :Pengelompokan berdasarkan kepadatan dan grid”, guna mencari kelemahan dan melakukan peluang untuk pengembangan algoritma sehingga algoritma ini dapat bekerja lebih baik, sedankan untuk pendalam ”CLIQUE: Subspace pengelompokan” berguna untuk mengetahui lebih jelas tentang cara kerja dan peran CLIQUE dalam pengelompokan subspace sehingga perlu dilakukan penyempurnaan agar algoritma ini dapat ditingkatkan ketelitian dalam menentukan pengelompokan terhadap data berskala besar, selanjutnya pengkajian terhadap “CLIQUE: Proses Generasi Calon Unit Padat” Guna mengetahui proses pencarian calon unit padat diperlukan untuk meningkatkan proses agar proses pencarian calon unit padat dapat bekerja lebih efektif.

Kedua: melakukan ”Analisa algoritma CLIQUE“ guna mengetahui keunggulan dan kelemahan sehingga algoritma tersebut dapat ditingkat dalam tahap prosesnya, sehingga algoritma yang diusulkan dapat bekerja dengan laju dan ketelitian tinggi, selanjutnya dilakukan penyempurnaan dengan menambahkan beberapa fungsi terhadap algoritma tersebut.

Ketiga : melakukan pengembangan terhadap algorima CLIQUE guna ditempelkan pada algoritma yang diusulkan “DENSITY-PRALLEL“, sehingga tahap-tahap proses untuk pengelompokan dapat disempurnakan sesuia dengan fungsi yang diingikan, sedangkan pengkajian teradap “Proses Generasi Calon Unit Padat” diperlukan untuk meningkatkan proses dan laju dari algoritma yang diusulkan, selanjutnya kajian ”Grid Adaptip: Efek grid pada kualitas pengelompokan” diperlukan untuk proses automatis terhadap proses pengelompokan sehingga algoritma ini dapat bekerja tanpa melakukan parameter masukan dari pengguna, yang disebut juga algoritma *”unsupervisi”*.

Keempat : kajian “Pemilihan Proses Perselarian” diperlukan untuk mengetahui kemampuan dan peluang algoritma ini agar dapat bekerja lebih efektif dalam proses perselarian, untuk pengkajian “Selari data” di perlukan untuk mengetahui proses selari terhadap data yang terdistribusi, mengetahui cara kerjanya dalam jejaringan yang diusulkan dengan tujuan proses ini digunakan untuk melakukan pendistribusian data ke semua kepemproses, sedangkan kajian terhadap ”Selari tugas” diperlukan untuk melengkapi proses selari sehingga tugas sama dapat dilakukan pada pemproses yang berbeda dengan data yang berbeda.

Kelima: proses “Analisa dan pemilihan perselarian pada :DENSITY- PARALLEL” diperlukan untuk pemilihan bentuk dan kaedah yang digunakan pada algoritma selari, sesuai dengan rancangan semula.

Keenam : tahap ”Algoritma selari yang diusulkan : DENSITY-PARALLEL” diperlukan untuk menjabarkan cara kerja detil dari algoritma yang diusulkan dan melakukan pengubahan dan penyempurnaan terhadap pada algoritma pengelompokan data yang ada, selanjutkan kajian lebih dalam terhadap fungsi algoritma yang diusulkan ”Algoritma: Membangun Calon Unit Padat” dimana algoritma ini dapat bekerja mencari calon unit padat pada proses selari dengan tingkat dan proses yang tinggi, sedangkan pendalaman “Algoritma: Mengenali Calon Unit Padat dan membentuk struktur data” guna menyempurnakan tahap algoritma yang diusulan dalam pengenalan calon unit padat dan membangun struktur data pengelompokan.

Ketujuh: tahap Eksperimen, diperlukan untuk melakukan percobaan terhadap algoritma yang diusulkan guna melihat hasil yang diharapkan, sesuai dengan tingkat penyempurna an yang dilakukan.

Terakhir: Hasil dan kesimpulan, diperlukan guna mendapatkan kesimpulan dari riset yang dibuat.

3.3 Asumsi Dan Pembatasan

Data pengelompokan secara selari pada algoritma pengelompokan perselarian yang akan dikembangkan dalam tesis ini perlu dilakukan pembatasan dan ruang lingkup pembahasan seperti berikut ini:

Pengkajian mencakup tentang kaedah pengelompokan data secara selari agar kaedah tersebut dapat diproses pada data yang berskala besar dan berdimensional tinggi dengan laju dan ketelitian yang tinggi, dalam hal ini penulis membuat batasan iaitu tentang penggunaan dan perluasan dari algoritma CLIQUE, algoritma yang dikembangkan dapat berkerja pada selari data dan selari proses laju dan ketilitian yang tinggi. Algoritma yang dikembangkan ini tidak memerlukan parameter masukan dari pengguna (unsupervisi algoritma). Sedangkan proses perselarian yang dilakukan dilingkungan jejaringan komputer windows.

3.4. Kajian Tesis : kaedah yang diusulkan

3.4.1. CLIQUE: Pengelompokan berdasarkan kepadatan dan grid

Pengelompokan berdasarkan kepadatan pendekatan berlaku suatu ciri-ciri pengelompokan lokal, di mana pengelompokan diunggulkan sebagai daerah dalam ruang data yang merupakan objek-objek padat dan yang dipisahkan oleh daerah objek kepadatan rendah (gaduh). Suatu cara umum untuk temukan daerah kepadatan tinggi dalam ruang data didasarkan pada kepadatan sel grid [A27]. Suatu histogram dibangun dengan penyekatan ruang data ke dalam sejumlah yang tidak tumpang tindih daerah manapun. Sel mengisi sejumlah objek pusat pengelompokan berpotensi dan batasan-batasan antara pengelompokan jatuh di dalam "lembah" pada histogram. Ukuran sel menentukan perhitungan dan mutu pengelompokan. Volume sel kecil akan memberi memberikan gungguan pada perkiraan kepadatan, sedangkan sel besar cenderung memperlancar perkiraan kepadatan. Wavecluster [A37] adalah suatu kepadatan dan grid berdasarkan pendekatan dengan melakukan pengubahan wavelet bentuk ruang corak. Penghitungan ini sangat efisien tetapi hanya dapat digunakan untuk data dimensional rendah. Wavecluster juga temukan uraian pengelompokan pada level resolusi berbeza dengan menerapkan multi-resolution wavelet ubah bentuk. Lebih lanjut, memerlukan suatu langkah pos proses dalam rangka menguraikan pengelompokan yang ditemukan.

3.4.1.1 CLIQUE: Subspace pengelompokan

Di dalam pengelompokan grid dan berdasarkan kepadatan dalam ruang subspace pendekatan ukuran grid sangat menentukan perhitungan dan mutu pengelompokan. CLIQUE, suatu kepadatan dan berdasarkan grid pendekatan untuk set data dimensional tinggi [A02,50B,50C], mendeteksi pengelompokan di dalam dimensional subspace sangat tinggi. Berdasarkan kepadatan pendekatan pengelompokan sebagai daerah kepadatan tinggi dibanding lingkungannya. Suatu cara umum menemukan daerah kepadatan tinggi di dalam ruang data didasarkan pada kepadatan sel grid itu [A27]. CLIQUE mengambil ukuran grid dan suatu ambang pintu kepadatan global untuk pengelompokan sebagai parameter masukan. Kompleksitas perhitungan dan mutu pengelompokan sangat tergantung pada parameter ini. Suatu histogram dibangun dengan penyekatan ruang data ke dalam suatu nomor yang tidak overlap dengan daerah dan kemudian memetakan data menunjuk masing-masing sel di grid. Persamaan interval panjang digunakan di dalam [A02,47,49,50] untuk mensekat masing-masing dimensi, menghasilkan volume sel seragam. Jumlah pada poin-poin di dalam sel dengan volume sel dapat digunakan untuk menentukan kepadatan sel.

Pengelompokan adalah perserikatan menghubungkan sel kepadatan tinggi. Dua sel k-dimensional dihubungkan jika mereka mempunyai suatu wajah umum di dalam ruang k-dimensional atau jika mereka dihubungkan oleh suatu sel umum. Menciptakan suatu histogram untuk menghitung pengisian poin-poin pada setiap unit adalah tidak mungkin dalam data dimensional tinggi. pengelompokan subspace lebih lanjut mempersulit masalah mengakibatkan ledakan data karena banyaknya subspaces bersifat exponen di dalam dimensi set data. Suatu pendekatan dari bawah ke atas menemukan unit padat dan menggabungkannya untuk temukan pengelompokan padat di dalam dimensional subspace lebih tinggi telah diusulkan di dalam CLIQUE [A02,45,50B,50C] . Algoritma ini serupa dengan algoritma secara teori [A05] yang digunakan di dalam perlombongan aturan asosiasi. Secara formal, bahwa kawasan Jika suatu "koleksi poin-poin S adalah suatu pengelompokan di dalam suatu ruang k-dimensional, kemudian S juga bagian dari suatu pengelompokan di dalam beberapa (k-1)-dimensi proyeksi ruang" [A02,45,50B,50C]. Masing-masing dimensi dibagi menjadi nomor seorang pemakai khusus pada interval, $\in$. Algoritma dimulai dengan menentukan unit 1-dimensional padat dengan pembuatan suatu tanda di atas data itu. Di dalam [A02,45,50B,50C] calon sel padat di dalam suatu k dimensi diperoleh dengan menggabungkan sel padat di dalam (k-1) dimensi yang berbagi dengan (k-2) dimensi pertama. Suatu tanda di atas data dibuat untuk mencari calon sel padat benar-benar aktual. Algoritma berhenti ketika calon sel padat tidak lagi dihasilkan. Di dalam [A02] calon unit padat adalah didasarkan pemotongan pada suatu teknik minimum uraian panjang untuk temukan unit padat hanya di dalam kaedah subspace. Bagaimanapun, seperti dicatat di dalam [A02] boleh mengakibatkan kehilangan beberapa unit padat di dalam pemotongan subspaces. Dalam rangka memelihara mutu pengelompokan yang tinggi kita tidak menggunakan teknik pembabatan ini di dalam implementasi DENSITY-PARALLEL yang kita usulkan

Kaedah untuk generasi calon unit padat diusulkan di dalam [A02,45,50B], tidak menyelidiki semua kemungkinan kombinasi unit padat dimensi lebih rendah. Sebagai contoh, pertimbangan dua unit padat 3-dimensional { a1, b7,c8} dan { b7, c8,d9} di mana (a, b, c, d) adalah bin

di dalam dimensi yang ditandai oleh penulisan subskrip. Ruang angka digambarkan oleh suatu pesanan dimensi { 1,...10}. Dengan mudah lihat bahwa dua hasil unit padat didalam calon unit padat 4-dimensi { a1,b7,c8,d9} yang mana tidak dibentuk oleh pendekatan di dalam [A02]. Seperti itu, di dalam pendekatan ini calon sel padat pada k dimensi, diperoleh dengan menggabungkan dua sel padat, yang diwakili oleh suatu dipesan (k -1) dimensi, dan mereka berbagi dengan (k -2) dimensi. Sekarang akan disajikan suatu analisa terperinci untuk memperoleh kemungkinan kesalahan yang dihasilkan dengan penggunaan Calon Unit Padat (CUP) teknik generasi yang diusulkan dalam [A02]

3.4.1.2 CLIQUE: Proses Generasi Calon Unit Padat

Calon unit padat di dalam dimensi k diperoleh dengan menggabungkan ke dua sel padat, diwakili oleh suatu pesan satuan $(k-1)$ dimensi, seperti itu bahwa mereka pertama berbagi untuk $(k-2)$ dimensi. kita akan melaksanakan suatu analisa kes yang sama dengan dulu dan mencari total jumlah calon unit padat, N_{CLIQUE}, dapat dihasilkan untuk suatu set data yang berisi pengelompokan dengan pemenuhan suatu subspace maksimum. Masalah ini dapat dipecahkan dengan menggunakan suatu teknik pengangkaan secara menyeluruh. Sama dengan dulu kita akan lihat masalah sebagai perpanjangan suatu urutan $k-1$ pada sebuah urutan k. Pertimbangan suatu urutan $k-1$ yang dibentuk dengan penempatan nomor i di dalam posisi i^{th}, $\{1,2,...,(k-1)\}$. Suatu urutan k dapat dibentuk dari urutan $k-1$ ini dengan pemilihan tiap satu bilangan bulat antara k dan d dan menempatkannya didalam posisi k^{th}. Dengan begitu urutan $d-(k-1)$ pada panjang k yang diberi subjujukan bersama-sama. Mengikuti bahwa dari satuan ini pada urutan $d-(k-1)$, masing-masing panjang $(k-1)$, dapat dikombinasikan keduanya membentuk suatu urutan k. Total seperti urutan *k* adalah ${}^{d-(k-1)}C_2$. Sekarang pertimbangan $k-1$ subjujukan $\{1,2,...,k\}$, dimana posisi $(k-1)^{th}$ berisi bilangan bulat k. Itu dapat dengan mudah dilihat bahwa kita mempunyai urutan $(d-k)$ masing-masing panjang *k* mempunyai subjujukan yang diberi bersama-sama. Dengan begitu satu urutan $(d-k)$ dapat dikombinasikan antar diri mereka untuk memberi suatu total jumlah urutan ${}^{d-k}C_2$ pada panjang *k*. Dalam suatu analisator ketika bertukar-tukar bilangan hadir di posisi $(k-1)^{th}$ dari $k-1$ untuk *d* dapat membentuk suatu total urutan panjangan k.

$$ {}^{d-(k-1)}C_2 + {}^{d-k}C_2 + ... + {}^{2}C_2 \qquad (3.2)$$

Sekarang dapat bertukar-tukar bilangan bulat dalam posisi $(k-2)^{nd}$ dan melaksanakan suatu analisa serupa. Jika bilang integer $k-1$ menduduki posisi $(k-2)^{nd}$ dan dapat bertukar bilangan bulat ke dalam posisi $k-1$ dari *k* kepada *d* dan dapat membentuk total dari calon unit padat (urutan dari panjang k).

$$ {}^{d-k}C_2 + {}^{d-(k+1)}C_2 + ... + {}^{2}C_2 \qquad (3.3)$$

Dengan cara yang sama jika k menduduki posisi $(k-2)^{nd}$ kita dapat membentuk suatu tambahan urutan panjang k.

$$ {}^{d-(k+1)}C_2 + {}^{d-(k+2)}C_2 + ... + {}^{2}C_2 \qquad (3.4)$$

Dengan begitu bermacam-macam bilangan bulat dalam posisi (*k*-2)nd dapat membentuk sebuah total urutan panjang k.

$$ \{{}^{d-k}C_2 + {}^{d-(k+1)}C_2 + ... + {}^{2}C_2\} + \{{}^{d-(k+1)}C_2 + {}^{d-(k+2)}C_2 + ... + {}^{2}C_2\} + ... + {}^{2}C_2 \qquad (3.5)$$

Dalam cara serupa kita bertukar-tukar bilangan bulat pada setiap posisi. Suatu kaedah pengangkaan ditemukan bahwa

$$ \begin{aligned} N_{CLIQUE} = {} & [{}^{D-(K-1)}C_2 + {}^{d-k}C_2 + ... + {}^{2}C_2] + \\ & [\{{}^{d-k}C_2 + {}^{d-(k+1)}C_2 + ... + {}^{2}C_2\} + \\ & \{{}^{d-(k+1)}C_2 + {}^{d-(k+2)}C_2 + ... + {}^{2}C_2\} + \\ & ... + \{{}^{2}C_{+}\}] + \end{aligned} \qquad (3.6)$$

dalam rangka menyederhanakan ungkapan di atas format tertutup dapat digunakan untuk mengevaluasi N_{CLIQUE}

$$ {}^{n}C_2 + {}^{n-1}C_2 + ... + {}^{3}C_2 + {}^{2}C_2 = \frac{n(n^2-1)}{6} \qquad (3.7)$$

Dengan begitu kita mempunyai

$$ N_{CLIQUE} = \sum_{n_{k-3}=2}^{n-(k-2)} ... \sum_{n2=2}^{n3-3} \sum_{n1=2}^{n2-2} \sum_{i=2}^{n1-1} \frac{i(i^2-1)}{6} $$

dimana $n_1, n_2, ..., n_{k-3}$ adalah sama dengan $d-(k-1)$.

Memberikan suatu nilai untuk *k* dan *d* kita dapat mengevaluasi nilai N_{CLIQUE} menggunakan suatu alat seperti mathematika. Hasil diatas dapat dilihat N_{CLIQUE} itu adalah banyaknya calon unit padat yang akan dibentuk dalam dimensi *k* untuk suatu set data yang mempunyai subspace pemenuhan maksimum oleh pendekatan dalam CLIQUE. Dan $N_{DENSITY-PARALLEL}$ adalah banyaknya calon unit padat yang akan dibentuk oleh algorima yang diusulkan: DENSITY-PARALLEL bahawa di dalam dimensi k untuk suatu set data. Serupa dengan itu kemungkinan kesalahan, P_e, tentang kehilangan pembentukan calon unit padat oleh pendekatan di dalam CLIQUE diberikan oleh

$$ P_e = 1 - \frac{N_{CLIQUE}}{N_{DENSITY-PARALLEL}} $$

dimensi set data. ,*d*, adalah 10 dan dimensi k di mana, akan cari calon unit padat 5 dan kita mempunyai N_{CLIQUE} untuk 210 dan $N_{DENSITY-PARALLEL}$ untuk 3150 dan kerananya P_e mengevaluasi ke 0,933. Kerana data besar dengan suatu subspace pemenuhan besar seperti itu kemungkinan kesalahan tinggi dapat diamati

3.4.2. Analisa algoritma CLIQUE

Di dalam diskusi yang diikuti Nup dijadikan Nombo/banyaknya Unit Padat di dalam dimensi k-1 yang berkombinasi untuk membentuk calon unit padat di dalam dimensi k. d jadikan dimensi data yang digunakan di dalam proses pengelompokan. Untuk mengevaluasi total jumlah calon unit padat yang boleh dihasilkan oleh algoritma DENSITY-PARALLEL dan diikuti oleh banyaknya CUPs yang dihasilkan. Seperti yang digunakan pendekatan pada [AGGR98]. Akhirnya kita akan mengkalkulasi kemungkinan kesalahan di dalam calon proses generasi unit padat yang digunakan algoritma di dalam [A02] dibandingkan dengan algoritma yang diusulkan:DENSITY-PARALLEL.

Masing-Masing unit padat di dalam dimensi k dengan sepenuhnya diwakili oleh k dimensi di mana unit padat hadir dan k indeks bin di dalam dimensi masing-masing mereka. Sebagai contoh, (1_1,3_2,4_5,7_2) menghadirkan suatu unit padat 4 dimensi di dalam dimensi (1,3,4,7) dan subskrip mereka menandai indeks bin di dalam dimensi

3.4.3. Kembangan algoritma CLIQUE pada DENSITY-PARALLEL

3.4.3.1. Proses Generasi Calon Unit Padat

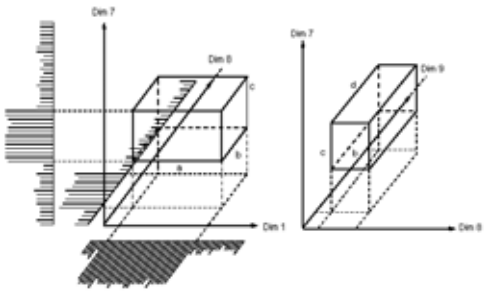
Calon unit padat di dalam k dimensi diperoleh dengan menggabungkan dua sel padat yang diwakili oleh suatu pesanan $(k-1)$ dimensi yang terdiri dari berbagai $(k-2)$ dimensi. Masing-masing unit padat perlu untuk dibandingkan dengan semua unit padat yang lain untuk mengidentifikasi CUPs, menghasilkan suatu algoritma $O(Nup^2)$. Dalam kes pengelompokan mempunyai suatu pemenuhan subspace besar, unit padat di dalam k dimensi terjadi di dalam tiap-tiap k dimensi subspace total ruang data pada dimensi d. Dalam rangka melaksanakan suatu analisa kes mari kita berasumsi bahwa data yang diberi mempunyai pengelompokan yang ditempelkan. Dalam semua subspaces menghasilkan unit padat $k-1$ dimensi di dalam ${}^{d}C_{k-1}$ subspaces. Pertimbangan masing-masing generasi dari calon tunggal unit padat dengan kombinasi dua unit padat. Dua unit padat dapat dikombinasikan bersama-sama jika mereka berbagi dengan k - 2 dimensi. Lebih lanjut jika lokasi-bin di dalam dimensi cocok. $k-1$ dimensi dari tiap unit padat dapat dilihat sebagai urutan $k-1$ bilangan bulat, semua kurang dari d. Dengan begitu pembuatan suatu k dimensi calon unit padat adalah setara dengan berkembang suatu $k-1$ urutan kepada suatu k urutan bilangan bulat. Jika d adalah masimum dimensional data, banyaknya $k-1$ subjujukan diberi oleh ${}^{d}C_{k-1}$. Masing-masing $k-1$ subjujukan dapat diperluas ke dalam suatu k urutan di dalam jalan $\{d-(k-1)\}$. Kerananya satuan tertentu jujukan $\{d-(k-1)\}$ masing-masing panjang k umumnya berisi suatu $k-1$ panjang subjujukan. Manapun dua set unit padat ini dapat dikombinasikan untuk membentuk suatu calon unit padat. Dengan begitu untuk menentukan $k-1$ subjujukan, kita dapat membentuk $\{d-(k-1)\}C_2$ calon unit padat. Jika $N_{density-parallel}$ menghadirkan total jumlah calon unit padat yang mungkin dapat dihasilkan.

$$N_{density-parallel} = {}^{d}C_{k-1} x^{\{d-(k-1)\}} C_2 \qquad (3.1)$$

seperti ${}^{d}C_{k-1}$ subjujukan masing-masing panjang $k-1$

3.4.3.2. Grid Adaptip

Perhitungan ongkos algoritma tergantung pada identifikasi dan perambatan ke dimensi lebih tinggi dari sel padat. Banyaknya calon sel padat menghasilkan dan diproses tergantung pada ukuran unit pada setiap dimensi. Seorang pemakai menggambarkan unit ukuran tetap tak mengindahkan distribusi data berkenaan dengan dimensi. Daerah jarang dan daerah padat adalah kedua-duanya mewakili dengan ukuran grid yang sama. Kerana daerah dengan pengelompokan ini mungkin memimpin ke arah suatu jumlah tinggi calon sel padat dan dengan suara gaduh data mungkin juga terdapat pada sel padat kerana kepadatan mereka adalah di atas ambang pintu tertentu. Pemakai lebih lanjut menggambarkan ambang pintu tidak mungkin mampu mengidentifikasi semua daerah padat yang ditempelkan. Gambar 3.1 memperlihatkan pembentukan suatu calon unit padat dalam dimensi yang menggunakan teknik dalam DENSITY-PARALLEL.



Gambar 3.1 : Pembentukan suatu calon unit padat $(k=4)$

Pertama menggambarkan beberapa notasi sebelum memperkenalkan konsep grid adaptip. $A=\{A_1, A_2, ..., A_d\}$ jadi satu set atribut dengan daerah $\{D_1, D_2, ..., D_d\}$ menjelaskan $S = A_1 x A_2 x ... x A_d$ suatu

d-dimensional ruang kwantitatip. $r = (r_1,...,r_d)$ yang dijadikan suatu d -dimensional masukan. Ruang S adalah disekat ke dalam suatu grid terdiri unit segi empat yang tidak tumpang tindih $C = c_{1k} x c_{2k} x ... x c_{dk}$, jadilah suatu sel (hyperrectangle) segi empat panjang, jika untuk semua $i \in \{1,...,d\}, c_{ik} \subseteq D_i$, $c_{ik} = [l_{ik}, u_{ik}]$ adalah interval dalam penyekatan pada A_i seperti itu bahwa $\mathrm{Y}_{allk}\ c_{ik} = D_i$.

Didalam [A02] masing-masing dimensi i disekat ke dalam $\in$ interval sama seperti itu $C_{ik} = \frac{D_i}{\in}$ untuk semua $k = 1,...,\in$. Suatu record $r = (r_1,...,r_d)$ terdapat di sel C, jika $l_{ik} \le r_i \le u_{ik}$ untuk semua c_{ik}. Suatu sel C adalah padat jika pecahan total data poin-poin terdapat di sel terlalu menjolok lebih besar (dengan beberapa faktor α) dibanding nilai yang mengharapkan jika data corak sama dibagi-bagikan kedalam ruang data. Suatu penyimpangan menjolok dari distribusi seragam dapat ditandai oleh suatu nilai α lebih besar dari 1.5. Algoritma 1 menguraikan langkah-langkah teknik grid adaptip. Daerah dari tiap dimensi adalah dibagi menjadi interval, masing-masing ukuran x. Maksimum untuk nilai histogram di dalam suatu jendela diambil untuk mencerminkan nilai jendela. .

Jendela bersebelahan nilai-nilai berbeda dengan kurang dari suatu persentase ambang pintu digabungkan bersama-sama untuk membentuk jendela lebih besar, memastikan bahwa membagi dimensi itu ke dalam lokasi ukuran variabel distribusi data. Pada gelombang segi-empat yang terbaik memenuhi data distribusi. Bagaimanapun dalam dimensi data yang bercorak sama dibagi-bagikan, ini mengakibatkan bin tunggal dan menandai adanya sangat sedikit kemungkinan kehadiran suatu pengelompokan. Dalam rangka menguji dimensi ini lebih lanjut kita merobek daerah ke dalam suatu jumlah partisi lebih kecil jumlah partisi dan mengumpulkan statistik untuk bin ini. Ini juga diijinkan untuk menetapkan suatu ambang pintu tinggi ketika dimensi ini adalah lebih sedikit nampaknya akan bagian dari suatu pengelompokan. Teknik ini sangat mengurangi waktu perhitungan ketika kita boleh membatasi derajat tingkat bagi bin dari bukan dimensi pengelompokan berperan untuk perhitungan. Dalam dimensi dengan ukuran variabel bin menetapkan suatu ambang pintu variabel untuk masing-masing bin dalam dimensi. Suatu bin dalam suatu dimensi adalah merupakan bagian dari suatu pengelompokan jika mempunyai suatu yang menjolok (oleh suatu faktor α) jumlah lebih besar dari poin-poin yang dibandingkan, itu telah mempunyai data bercorak sama dibagi-bagikan dalam dimensi. Dengan begitu untuk suatu ukuran bin adalah suatu ukuran dimensi D_i kita menetapkan ambang pintunya untuk $\frac{\alpha a N}{D_i}$, dimana N adalah total jumlah poin data. Suatu nilai α lebih besar dari 1.5 telah bekerja baik dalam eksperimen.

3.4.3.3. Efek grid pada kualitas pengelompokan

Gambar 3.2(a) menggambarkan grid yang seragam menggunakan dalam CLIQUE. Ini adalah bukan kenal distribusi data dan menghasilkan banyak lagi calon unit padat yang lain untuk memproses pada masing-masing melangkah dibanding suatu grid adaptip dalam Gambar 3.2(b). CLIQUE memerlukan suatu tahap praproses untuk menghasilkan
panjangnya uraian minimal pengelompokan untuk membuat pengertian pengelompokan lebih bersedia menerima masukan pengguna akhir. Ini adalah dipecahkan dengan penggunaan suatu algoritma CLIQUE meliput yang ditemukan grid dalam pengelompokan segiempat panjang maksimal menyediakan pemenuhan. Sejak itu suatu perkiraan tentang pengelompokan lebih lanjut menambah kompleksitas dan mengurangi ketepatan yang dilaporkan

Pada sisi lain kerana DENSITY-PARALLEL menggunakan batasan-batasan grid adaptip pengelompokan didefenisikan adalah FND (Format Normal Disjungsi) minimal yang menyatakan perlawanan ungkapan format normal dan melaporkan batasan-batasan pengelompokan yang jauh dengan teliti. Gambar 3.3 menunjukan suatu pengelompokan didefenisikan dalam dua dimensi.

3.4.4. Pemilihan Proses Perselarian

3.4.4.1. Selari data

Masing-masing pengolah dimulai dengan membangun suatu histogram untuk semua dimensi dengan data ukuran $\frac{N}{p}$ dimana N adalah total jumlah record data dan p banyaknya pengolah. Selama proses data dibaca dari cakera bersama dan menulis kepada cakera lokal dengan demikian data yang diakses berikut dapat dengan jalur lebar dan lebih cepat. Pengurangan komunikasi primitif dengan penjumlahan sebagai peoperasinya mengumpulkan histogram yang global pada pengolah masing-masing. Dengan suatu vektor ukuran m pada pengolah masing-masing dan suatu persekutuan operasi biner. Mengurangi operasi menghitungan suatu garis vektor hasil ukuran m dan menyimpannya pada tiap-tiap pengolah. Elemen i^{th} hasil vektor adalah hasil kombinasi elemen i^{th} pada garis vektor di semua pengolah yang menggunakan persekutuan operasi biner. Masing-masing pengolah sekarang menentukan interval terbatas yang mudah suai untuk tiap-tiap dimensi dan memperbaiki ambang pintu dan ukuran lokasi bin untuk tiap-tiap bin seperti dilakukan pada algoritma 1. Masing-Masing bin yang ditemukan dianggap sebagai suatu calon unit padat. Calon unit padat didiami oleh suatu tanda pada data lokal yang mana dibaca pada kumpulan record B. Memungkinkan algoritma dapat diterapkan untuk ke luar dari dapat data inti. Karena pengolah masing-masing hanya mengakses data lokal, pada ukuran $\frac{N}{p}$ selari data dapat diperoleh. Pengurangan operasi mendapatkan hitungan

histogram calon unit padat global pada semua pengolah. Ini diikuti oleh algoritma selari tugas dengan mengidentifikasi unit yang padat dan pembentukan struktur data.

3.4.4.2. Selari tugas

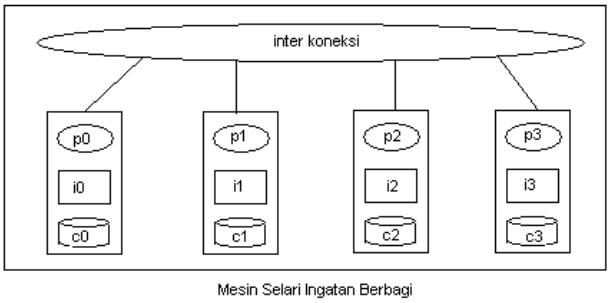
Tugas mencari calon unit padat dan mengidentifikasi unit padat di antara calon unit padat dibagi di antara pengolah, pada masing-masing pengolah mendapatkan suatu jumlah pekerjaan yang sama. Optimal tugas penyekatan adalah penting dalam rangka memastikan bahwa pengolah tidak menantikan pengolah yang lain sampai selesai tugasnya. Setelah penyelesaian pekerjaan yang ada pengolah merobah pesan dalam rangka memperoleh informasi yang global. Keuntungan selari tugas hanya ketika waktu perhitungan yang besar jauh dibandingkan biaya komunikasi. Masing-masing calon unit padat (CUP) dan unit padat yang sama, didalam d^{th}-dimensi dengan sepenuhnya ditetapkan oleh d-dimensi unit dan bersesuaian indek d lokasi mereka. Dalam implementasinya disiapkan informasi dalam wujud suatu array bait . Satu array untuk indeks bin dari semua CUPs dan satu untuk dimensi CUP. Masing-masing array yang sama digunakan untuk penyimpanan dimensi dan indeks bin unit padat. Dengan penyimpanan informasi dalam wujud suatu larik lurus bait tidak hanya mengoptimalkan ruang tetapi juga memperoleh kemudahan untuk berkomunikasi. Ini komunikasi informasi dalam langkah tunggal dengan penggunaan penyangga pesan jauh lebih kecil.

3.4.5. Analisa dan pemilihan perselarian pada : DENSITY-PARALLEL

Pertama akan diperkenalkan suatu perumusan pengukuran algoritma pengelompokan selari subspace yang mengunakan selari data dan tugas selari. Dasar mesin selari adalah arsitektur "shared-nothing". Beberapa unit pemproses ingatan cakera dipasang pada suatu jaringan komunikasi seperti ditunjukkan dalam Gambar 4.1. Program dioperasikan di kaedah SPMD (Single Program Multiple Data/ Program Tunggal Data Terbagi), dimana program yang sama beroperasi pada berbagai pengolah hanyalah penggunaan membagi-bagikan data penugasan kepada pemproses. Ini memimpin ke arah suatu mekanisme data selari alami. Tugas selari dicapai oleh bagian tugas yang ada menugaskan ke masing-masing pengolah. Pengolah mengkomunikasikan dengan pertukaran pesan. Dalam arsitektur selari menggunakan paradigma ini misalnya (Pentium 4) komunikasi kependaman yang lebih dari suatu order penting akan memerlukan waktu perhitungan yang lebih besar. Waktu komunikasi terdiri atas suatu komponen susunan koneksi dan kata pesan kependaman terdahulu disebut komponen yang lebih besar.

3.4.6. Algoritma selari yang diusulkan : DENSITY-PARALLEL

DENSITY-PARALLEL adalah suatu algoritma scalable dan selari berbasis cakera yang mampu menangani data besar dengan sejumlah dimensi besar. Algoritma dapat juga



Gambar 4.1 : Arsitektur "shared-nothing" P:Pemproses, I:Ingatan, C:Cakera

beroperasi dipengolah tunggal di mana langkah-langkah komunikasi akan diabaikan. Algoritma 2 menunjukkan langkah-langkah dalam algoritmanya. Berdasar pada Pentium-4 masing-masing pengolah membaca sebagian data dari suatu cakera bersama pada awalnya dan menyimpannya pada cakera lokal. Luas bidang yang dilihat oleh suatu pengolah dari suatu akses I/O dari cakera lokal jauh lebih tinggi dibanding suatu akses bagi suatu cakera bersama.

Algorithm 2: DENSITY-PARALLEL

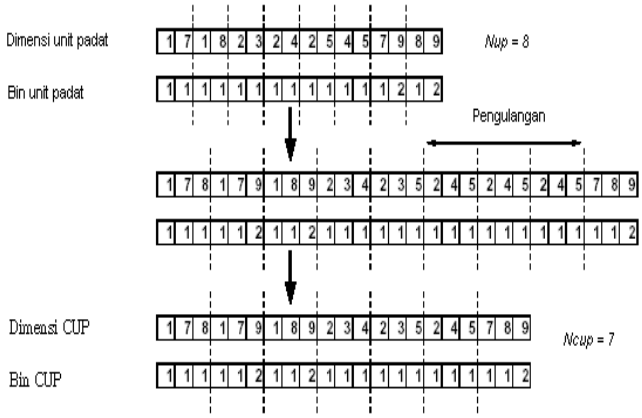
N *- jumlah record;*
p *- jumlah pemproses;*
d *- dimensi data*
A_i *-* i^{th} *atribut* $i \in d$ *;*
B *- jumlah record yang cocok di ingatan penyangga dialokasikan pada pemproses masing-masing*

/ Masing pemproses membaca data* $\frac{N}{p}$ *dari cakera lokalnya*/*
Pada setiap pemproses
Baca $\frac{N}{p}$ *tumpukan record B pada cakera lokal dan bangun suatu histogram pada setiap dimensi* $A_i, i \in (1,...,d)$
Kurangi komunikasi untuk mendapatkan histogram global
Tentukan interval mudah suai menggunakan histogram itu pada setiap dimensi $A_i, i \in d$ *dan juga tingkatan ambang pintu*
Set calon unit padat kepada lokasi penemuan pada setiap dimensi
Set dimensional sekarang k *ke 1*
Kerjakan selagi(tidak ada unit padat ditemukan)
Jika ($k > 1$*)*
Cari-calon-unit-padat();
Baca $\frac{N}{p}$ *tumpukan record B pada cakera local dan untuk tiap-tiap populasi record calon unit padat*
Kurangi komunikasi untuk mendapatkan populasi calon unit padat global

Identifikasi-unit-padat();
Daftarkan yang bukan unit padat dengan cetak struktur data pada pemproses utama.
Membangun-struktur-data-unit-padat();
Jika(Pemproses utama)
Cetak-pengelompokan();
Berakhir

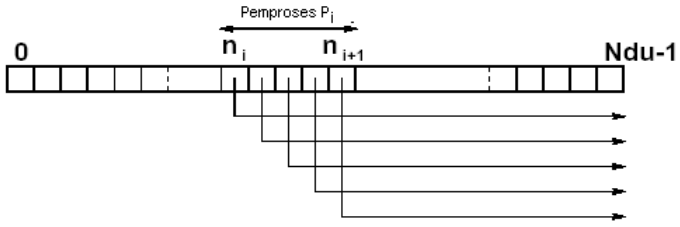
DENSITY-PARALLEL berisi sebagian besar langkah-langkah berikut ini, Unit calon padat dalam dimensi k dibangun dengan kombinasi unit dimensi padat $k-1$. bahwa mereka berbagi ke dimensi $k-2$. Algoritma selari *Cari-calon-unit-padat()* dipengaruhi dalam memproses ini. Algoritma menghabiskan banyak dari waktunya dalam membuat suatu tanda atas data dan mengenali unit unit padat di antara calon unit padat yang terbentuk dalam tiap-tiap dimensi, Pengulangan tanda atas data perlu untuk dilaksanakan ketika kemajuan algoritma membangun unit padat dari dimensi lebih tinggi. Setelah pencarian penghitungan histogram calon unit padat dalam dimensi, unit padat yang diidentifikasi dan unit padat struktur data dibangun untuk dimensi lebih tinggi berikutnya. Algoritma *Identifikasi-unit-padat()* dan *Membangun-struktur-data-unit-padat()* akan dijelaskan secara detil. Setelah memperkenalkan kedua-duanya data selari untuk IO tahap intensive membangun hitungan histogram calon unit padat dan selari tugas untuk semua tugas dalam algoritma. algoritma akan berakhir ketika unit padat tidak lagi pada pengelompokan, dan akhirnya dicetak oleh pemproses utama di ujung program

3.4.6.1. Algoritma : Membangun Calon Unit Padat



Gambar 4.2: Membangun calon unit padat (Dimensi k = 3)

Gambar 4.2 adalah proses membangun calon unit padat dalam 3 dimensi untuk suatu set data 10 dimensi. Calon unit padat dalam manapun dimensi k dibentuk dengan kombinasi unit padat dimensi $k-1$ seperti itu bahwa mereka berbagi beberapa $k-2$ dimensi. Langkah-langkah dari algoritma ini ditunjukkan dalam algoritma 3. Banyaknya unit padat ditandai oleh Nup dan banyaknya CUPs oleh $Ncup$. Adalah mudah untuk dilihat bahwa masing-masing unit padat perlu untuk diuji dengan semua unit padat yang lain untuk membentuk calon unit padat dan ini akan mendorong kearah suatu algoritma $O(Nup^2)$. Ini akan berjumlah suatu perhitungan sangat besar ketika Nup adalah besar. Sesuatu dapat menghasilkan CUPs di dalam selari dan kecepatan keseluruhan proses. Sekarang dapat memperoleh formula mencapai optimal tugas penyekat memproses calon generasi unit padat dengan hasil kecepatan yang bagus. k jadi dimensi di mana pembangunan calon unit padat. Jika Nup adalah jumlah unit padat dengan dapat mudah lihat bahwa total nilai perhitungan dilakukan adalah $\frac{Nup(Nup+1)}{2}$. Ketika setiap pengolah bekerja pada seluruh unit padat. $n_i, n_2, ..., n_{p-1}$ angka nyata seperti $0 < n_1 < n_2 < ... < n_{p-1} < Nup$ dan bagi Nup unit padat pada bagian p seperti setiap bagian pada array unit padat dapat di proses oleh satu p pengolah. Bagian Nup ke dalam bagian p sehingga setiap pengolah mengerjakan suatu jumlah perkerjaan sama, sama ke $\frac{Nup(Nup+1)}{2p}$. Jika pengolah ranking i beroperasi dalam daerah antara n_i dan n_{i+1} membandingkan unit padat dari n_i ke n_{i+1} dengan semua unit



Gambar 4.3 : Pembentukan selari pada calon unit padat

padat lain lebih besar dari n_i sebagaimana ditunjukkan dalam gambar 4.3.

$$Nup * (n_{i+1} - n_i) - (\sum_{j=n_i}^{n_{i+1}-1} j) = \frac{Nup(Nup+1)}{2p} \qquad (4.1)$$

Mecahkan $p-1$ pelelaran penyamaan untuk $n_1, ..., n_{p-1}$ yang mulai dari n_1 akan dapat memperoleh suatu optimal tugas partisi untuk menemukan calon unit padat. Setelah memperoleh solusi untuk n_i, nilai n_{i+1} dapat diperoleh dengan pemecahan persamaan quadrat diatas. CUPs yang dihasilkan oleh pengolah dikomunikasikan kepada pengolah utama yang menggabungkan dimensi CUP dan lokasi-bin array dalam ranking memesan pengolah. Informasi ini disiarkan kepada semua pengolah.

Algoritma 3: Cari-calon-unit-padat()

$Ncup$ *- Jumlah calon unit padat ;*
CUP – Calon unit padat
Nup *- Jumlah unit padat ;*
p *– Jumlah pemproses ;*
τ *– tetapan*

Pada setiap pemproses
Jika ($Nup > \tau$ *)*
Cari awal dan akhir indek pada partisi array unit padat yang harus diproses.
Bangun CUPs dari partisi unit padat
Kirim informasi dari hitungan lokal CUP, Dimensi CUP dan Bin pada pemproses utama
Kirim informasi Unit padat yang tidak dapat di kombinasikan pada pemproses utama
Jika(Pemproses utama)
Terima jumlah lokal CUP, Dimensi CUP, index Bin CUP untuk seluruh pemproses
Hitung total jumlah CUP, $Ncup$ *dan gambungkan dimensi CUP dan indek bin CUP dalam rangkingnya*
Sebarkan $Ncup$ *dan gabungkan dimensi CUP dan information bin*
Terima unit padat yang tidak boleh dikombinasikan dan perbaharui struktur data yang digunakan untuk mencetak pengelompokan
Penghapusan-pengulangan-CUPs()
Selain
Bangun calon unit padat pada k *dimensi dari seluruh unit padat pada* $(k-1)$ *dimensi*
Jika (Pemproses utama)
Daftarkan unit padat yang tidak boleh dikombinasikan dengan struktur data yang digunakan untuk pencetak pengelompokan.
Penghapusan-pengulangan-CUPs()
Berakhir

Unit padat yang tidak boleh dikombinasikan dengan unit padat lain dicatatkan di pengolah utama sebagai kelompok potensial dalam dimensi $k-1$. Calon unit padat dihasilkan dalam selari hanya ketika masing-masing pengolah dijamin untuk mempunyai suatu minimal jumlah pekerjaan. Jika $Ncup$ kurang dari suatu tetapan τ, CUPs dihasilkan oleh semua pengolah dengan pengolahan semua unit padat

Dari Gambar 4.2 bahwa proses generasi CUP boleh mendorong kearah dibentuk calon unit padat serupa. Untuk itu CUPs harus mengidentifikasi ulangi dan mempertahankan hanya unsur-unsur yang unik. Penghapusan dari CUPs serupa tidaklah hanya diperlukan tetapi juga sangat mengurangi waktu tugas selari bagian dari algoritma. Suatu keperluan harus membandingkan masing-masing CUP dengan semua CUP yang lain untuk mengidentifikasi unsur-unsur yang diulangi dengan menghasilkan suatu $O(Ncup^2)$ algoritma kompleksitas. Dengan begitu ketika $Ncup$ adalah besar kita mengidentifikasi ulang calon unit padat di dalam salari. Ini serupa kepada pembuatan calon unit padat pada selari tugas yang diperoleh pemecahan atas pelelaran $(p-1)$ penyamaan dan dengan Nup digantikan oleh $Ncup$ .Lebih lanjut jika banyak calon unit padat unik yang dikenali masih besar $(>\tau)$ dibangun struktur data CUPs dalam selari dan akhirnya penukaran pesan untuk memperoleh informasi global. Langkah-langkah algoritma dapat dilihat pada algoritma 4.

Algoritma 4 : Penghapusan-pengulangan-CUPs()

$Nulang$ *- Pengulangan CUP*
Setiap pemproses
Jika ($Ncup > \tau$ *)*
Temukan awal dan akhir indek partisi pada array CUP agar di proses.
Identifikasi ulang CUPs dalam keseluruhan array CUP bandingkan dengan bagian array CUPs.
Kurangi komunikasi untuk memperoleh informasi global yang diulangi CUPs
Set nilai $Nulang$ *dan* $Ncup = Ncup - Nulang$
Bangun-cup-dengan-elemen-unik();
Selain
Identifikasi yang diulangi CUPs
Set nilai $Nulang$ *dan* $Ncup = Ncup - Nulang$
Bangun-cup-dengan-elemen-unik();
Berakhir
Bangun-cup-dengan-elemen-unik();
{
Jika ($Ncup > \tau$ *)*
Bagi $Ncup$ *dengan* p *dan temukan awal dan akhir indek aray CDU yang diproses.*
Bangun dimensi CUP $(\frac{1}{p})^{th}$ *dan hilangkan elemen CUP yang berulang.*
Kirimkan format informasi CUP $(\frac{1}{p})^{th}$ *ke pengolah utama.*
Jika (Pengolah utama)

Terima informasi CUP $(\frac{1}{p})^{th}$ *dari seluruh pemproses.*

Gabungkan mereka dalam ranking ordernya

Sebarkan informasi global CUP ke semua pengolah

Selain

Bangun struktur data CUP pada dimensi CUP dan hilangkan bin unsur-unsur CUP yang berulang.

}

3.4.6.2. Algoritma: Mengenali calon unit padat dan membentuk struktur data

Calon unit padat harus dicuba untuk membuktikannya benar-benar padat. Masing-masing pengolah memilih $\frac{1^{th}}{p}$ calon unit padat untuk menentukan unit padat. Hitung histogram dari setiap CUP bandingkan dengan ambang pintu dari seluruh bin yang dibentuk CUP. Penghitungan lokal dari unit padat yang ditemui adalah dipelihara pada setiap pemproses. Suatu pengurangan komunikasi sehingga semua pengolah mempunyai informasi unit padat dari set calon unit padat.

Pengurangan komunikasi yang lain dilakukan untuk memperoleh penghitungan global banyaknya unit padat (Nup) pada semua pengolah. Jika jumlah calon unit padat lebih bekurang dari τ, masing-masing pengolah bekerja pada seluruh calon unit padat untuk menentukan unit padat. Algoritma 5 menampilkan langkah yang terlibat. Sama dengan diatas jika banyaknya unit padat adalah lebih besar dari τ maka struktur data unit padat dibangun secara selari, Ini penting untuk dicatat bahwa harus secara hati-hati membagi tugas dalam membangun struktur data ketika unit padat tidak dibagi-bagikan. Suatu pencarian linear diatas array unit padat diperlukan untuk menentukan awal dan akhir indek antara masing-masing pengolah untuk membagi tugas yang sama. Struktur data pada indek bin unit padat dan dimensinya dibangun secara selari kini digabung dengan pengurangan operasi komunikasi.Langkah-langkah algoritma ditunjukan dalam algoritma 6. Algoritma kemudian memproses dimensi lebih tinggi dan mulai membangun calon unit padat berakhir ketika tidak ada calon unit padat lagi.

Setelah proses pendeksian pengelompokan telah diselesai, pemproses utama memproses semua daftar masukan dalam struktur data untuk mencetak pengelompokan. Pengelompokan yang sama dimensi tinggi dihapuskan dan hanya pengelompokan dimensi unik paling tinggi diperkenalkan pada pemakai akhir. Ini peningkatan penggunaan algoritma jauh lebih besar .

3.4.7. Analisa dan rangkuman

k menghadirkan dimensi tinggi pada segala unit padat dalam set data. Lama operasi algoritma adalah eksponensial k. Ini adalah fakta jika sebuat sel padat yang ada pada dimensi k kemudian semua diproyeksi ke dalam sub k-dim

Algoritma 5: Identifikasi-unit-padat()

Pada setiap pemproses

Jika ($Ncup > \tau$*)*

Bagi $Ncup$ *dengan* p *.*

Temukan awal dan akhir indek partisi pada array CUP yang diproses.

Untuk setiap CUP dalam partisinya adalah array, bandingkan histogram CUP dengan ambang pintu Bin yang membentuk CUP

Tentukan CUP adalah padat atau bukan.

Pelihara suatu hitungan lokal banyaknya unit padatyang dideteksi.

Kurangi komunikasi untuk mendapatkan informasi unit padat dari semua CUPs diikuti yang lain.

Kurangi komunikasi untuk mendapatkan total jumlah unit padat Nup

Selain

Untuk seluruh CUPs, $Ncup$*, dibandingkan penghitungan histogram* CUP *dengan ambang pintu bin* CUP

Tentukan jika CUP adalah padat atau bukan

Temukan total jumlah untuk padat Nup *.*

Berakhir

Algoritma 6 : Membangun-struktur-data-unit-padat()

Pada setiap pemproses

Jika ($Nup > \tau$*)*

Bagi Nup *dengan* p

Temukan awal dan akhir indek dari array CUP yang diproses.

Bangun hubungan struktur data dengan dimensi dan indek bin unit padat dari bagian pada array CUP.

Kurangi komunikasi untuk mendapatkan informasi global struktur data unit padat.

Selain

Untuk seluruh CUPs, Bangun hubungan struktur data dengan dimensi dan indek bin unit padat

Berakhir.

mensi $O(2^k)$ kombinasinya adalah juga padat. Dengan begitu harus melihat kemungkinan pengelompokan dalam semua subspace diantara pengelompokan k dimensi.

Bagaimanapun dengan penggunaan dari *grid* mudah suai banyaknya calon unit padat sangat dikurangi dan dengan begitu memungkin DENSISTY-PARALLEL untuk mengelupas lebih dalam dimensional pada set data dan ukuran set data. α adalah konstanta untuk komunikasi dan

S adalah ukuran penukaran pesan antar pengolah dan N total jumlah record. Juga B banyaknya record yang ada dalam mengingat penyangga atas pengolah masing-masing dan γ adalah waktu akses I/O untuk satu blok B record dari lokal cakera. Komplesitas waktu penghitungan algoritma adalah $O(c^k)$, dimana c adalah konstanta. Total waktu I/O masing-masing pengolah adalah $O(\frac{N}{pB}k\gamma)$, seperti setiap pengolah harus membaca hanya $\frac{N}{p}$ bagian dari data dalam tumpukan B record. Faktor k adalah kaitan danta k diperlukan atas pangkalan data sebelum algoritma berakhir. Waktu komunikasi adalah $O(\alpha Spk)$. Total kompleksitas waktu algoritma adalah $O(c^k + \frac{N}{pB}k\gamma + \alpha Spk)$. Waktu operasi pengolah tunggal dengan mudah diperoleh dengan mengganti p =1 dan S =0, seperti tidak ada komunikasi.

DENSITY-PARALLEL adalah algoritma pengelompokan yang tidak diawasi. Pengelompokan ditemukan oleh algoritma tergantung dua parameter α dan $\beta . \alpha$ ditandainya besar penyimpangan nilai histogram yang tersebar purata. Suatu nilai lebih besar dari 1.5 telah di terima untuk menjadi penyimpangan cukup untuk dipertimbangkan sekali dalam bidang statistika dan perlombongan. Menemukan pengelompokan dengan nilai tinggi pada pengelompokan dalam set data jadi lebih dominan dibanding dengan yang lainnya dalam kaitan dengan banyaknya point data yang terdapat dalam pengelompokan. Karenanya memilih suatu nilai yang pantas adalah secara langsung. Parameter β adalah pengendalian proses penemuan *grid* mudah suai. Pengendalian β banyaknya bin yang dibentuk pada setiap dimensi. Suatu nilai rendah hasil β mengakibatkan penggabungan bin berdekatan yang mempunyai nilai histogram hampir serupa.Bagaimanapun nilai histogram pada bin yang berdekatan jarang yang sama. Dengan begitu nilai rendah hasil dalam sejumlah besar pada bin di setiap dimensi. Ini mengakibatkan suatu waktu lebih besar untuk penghitungan tetap akan menghasilan pengelompokan berkualitas lebih baik. Nilai tingi mengakibatkan penggabungan semua bin ditentukan dalam dimensi dan akan menghasilkan pengelompkan mutu rendah. Algoritma ini bukanlah yang sensistif dalam penilaian β.

Pengelompokan selari bermutu tinggi ditemukan secara efesien oleh DENSITY-PARALEL ketika terlalu rendah atau terlalu tinggi nilai b dihindarkan. Suatu nilai β sekitar 25% sampai 75% telah dapat bekerja baik pada percubaan ini secara rinci. Hasil laporan dalam kertas adalah sama dengan β sepadan dengan 35%.

5. Rujukan

[01] *Crawford, J., Crawford, F. "Data Mining in a Sciencetific Environment and Management Information". Autralia Menai Nsw.*

[02] *Eisen, M. (1999)."Cluster and Tree View". Manual software*

[03] *Epter, S., Krishanamoorthy, M., Zaki, M."Clusterability Detection and Initial Seed Selection in Large Data Sets".*

[04] *Faber, V." Clustering and the Continuous k-meands Algorithm".*

[05] *Guralnik, V., Karypis,G (2001)."A Scalable Algorithm of Clustering Sequential Data".*

[06] Han, J., Kember, M."Data Mining Concept And Techniques" and Intelligent Database Systems Research Lab School of Computing Science Simon Fraser University". Canada.

[07] *Han, E.H.S., Kumar, G.K.V., Mobasher, B."Clustering in a high dimensional space using hypergraph models".*

[08] *Han, E.H.S., Kumar, G.K.V., Mobasher, B. "Hypergraph Based Clustering in High Dimensional data sets a summary of results".*

[09] *Halkidi, M., Batistakis, Y., Vazirgiannis, M."On Clustering Validation Techniques".*

[10] *Huang, Z. "A Fast Clustering Algorithm to Cluster very large categorical Data Sets in Data Mining". Cooperative Research Centre for advanced computational systems. Australia.*

[11] I Made Wiryana,Ssi,Skom,Msc.(1995). "Penggunaan metoda soft computing untuk aplikasi bisnis".

[12] *Kreuseler, M., Nocke,T., Schumann, H." Integration of Cluster Analysis and Visualization Techniques for Visual Data Analysis".*

[13] *Peuquet, D.J and Guo, D."Mining Spatial Data Using An Interactive Rule-Based Approach".*

[14] *San jose,C.A.(2003)."Survey of clustering data mining techniques".*

[15] *Simon, U., Berndtgen, M." Wave Stat ClusterAnalysis of Image Data And Wavelet Coefficients".*

[16] *Sui, Z., Yang, Q., Zhang, H., Xu, X., Hu, Y."Correlation Based Document Clustering Using Web Log".*

[17] *Vesanto, J and Alhoniemi, E. "Clustering of the Self-Organizing Map".*

[18] *Wooley, C., Bridges, S., Hodges,J and Skjellum, A." Scaling the data mining step in knowledge discovery using oceanographic data".*

[19] Yi-Chin Lee.(2002)."Data Mining A tutorial".

[20] *Zhao, Y., Karypis, G." Clustering in life Sciences".*

[21] Williams, G., Altas, I., Bakin, S., Christen, P., Hegland, M., Marquez, A., Milne, P., Nagappan R, and Roberts, S.(2001)." The Integrated Delivery of Large-Scale Data Mining".

[22] Han, J., Chiang, J.Y., Chee, S., Chen, J.(2000)." A System for Data Mining in Relational Database and Data Warehouses". DBMINER

[25] Muntz, R., Potkonjak, M., Slijepcevic, S., Wang, W." Application to Client-Side Caching in Derect TV". (2000).Industrial Sponsor Hughes Aircraft Corp,2000

[26] Rocke, D.M. and Jian." With applications to Sky Survey Data".(1999). University of California; Davis.

[27] Johannes, V.J., Raghu ,R., Ramakrishnan. (1999). " Mining Very Large Databases ". University of Wisconsin Madison

[28] Motwani, R and Ullman, J.D.(2001)." Knowledge Discovery Through Large-Scale Data Mining"

[29] Edelstein,H (Two Crows Corporation).(2000) Mining large database – a case study

[30] ZHAO, F.(1999)." Intelligent Simulation Tools for Mining Large Scientic Data Sets.", 1999

[31] Choudhury, R., Nair, P.B and Keane, A.J.(2002)." A Data Parallel Approach for Large-Scale Gaussian Process Modeling", 2002.

[32] Rome, J.(2002)."Data Mining Data Description: Requirements for a Large Scale Data Mining Application".

[33] Zaiane, O.R., Han,J.(2000)."Mining Recurrent Item in Multimedia with Progressive Resolution Refinement."

[34] Moirhomme, M.(1999)."Multimedia support for complex multidimensional data mining", 1999

[35] Madria, S., Bhowmick, S.S., (w-k eng e.p.lim; 1998)."Research Issues in web datamining", 1998

[36] Model, C.(1998)."Generalization-based Data Mining in Object-Oriented Database using an object Cube Model", 1998

[37] Han, J.(1999)." Data Mining Techniques".

[38] Ye, X., Keane, J.A.(1998).".Mining Association Rules in Temporal Database".

[39] Goebel, M., Gruenwald, L.(june 1999)."A Survey of Data Mining and Knowledge Dicovery Software Tools".

[40] Chen, M.S., Han, J.(19996)."Data Mining: An Overview from a Database Perspective".19996

[41] Hansson, J.(1999)."Issues in Active Real-Time Database; Michael Berndtsson"

[42] Tinoco, L.C.(1996)."Online Evaluation in WWW-based CoursewareThe QUIZIT System".

[43] Keim, D.A., Kriegel, H.P, Seidl, T.(1994)."Supporting Data Mining of Large Databases by Visual Feedback Queries". University of Munich; Leopoldstr. 11B, D-80802 Munich, Germany.

[44] (2000) "Inteligent Assistance for the Data Mining Process An Antoloty_based Approach".

[45] Stühlinger1, W, Hogl, O, Stoyan, H, and Müller, M.(2000) ."Intelligent Data Mining for Medical Quality Management".

[46] Taniar, D and Smith,K.(1999)."Module 1a Introduction to Parallel Data Mining".

[47] V.Fahar. [1999], "Clustering and The Continous K-means Algorithms",

[48] Szymkawiak, J.Larsen. [1999], "Hirarchical Clustering For Data Mining"

[49] J.Grabmear, A Rodolph [1998], "Techiniques of Clusterring Algorithms in Data Mining", Desc, 1998.

[50] K.wagstaff & C.Cardie (2001), "Constrained K-Means Clustring With Background Knowledge", 2001.

[51] S.Chatterjee & J. Prins. (2003). "Parallel and Distributed Computing PRAM Algorithms", 2003.

[52] Gadomski, A.M, Balducelli, C, Bologna, S and DiCostanzo, G. (2000)."Integrated Parallel Bottom-up and Top-down Approach to the Development of Agent-based Intelligent DSSs for Emergency Management".