Analisis Sub-Graph Query pada Jaringan Komunikasi Email dengan Pendekatan GraphREL

GraphREL: A Decomposition-Based and Selectivity-Aware Relational Framework for Processing Sub-graph Queries

Ludovica Gorganusa (1103080042)
Fakultas Teknik Informatika, Telkom University
gorganusa@gmail.com

Abstrak **-** Graph database adalah alat pemodelan data yang bersifat sederhana sampai kompleks. Dengan berisikan node dan edge, suatu data dapat dimodelkan untuk memudahkan analisis suatu proses dalam bentuk query. Graph database lebih unggul daripada relational database karena bisa menangani data yang tidak terstruktur dan semi terstruktur. Pemodelan data kedalam bentuk graph bisa bermacam-macam. Misalnya saja kasus email, yang sifatnya dalam skala besar dan dinamis.

Satu alamat email bisa menampung ribuan email dari ratusan pengguna email dengan alamat email berbeda. Tentu database yang disajikan akan memakan banyak tabel dengan atribut yang sama, hanya isinya yang berbeda. Untuk satu alamat tujuan perlu dibentuk satu tabel baru yang menampung data email komunikasi antar pengguna. Lain halnya dengan menggunakan pemodelan dengan graph database.

Dari kasus tersebut memerlukan pendekatan sebuah framework relational yang berbasis dekomposisi dan Selektivitas-Aware untuk pengolahan sub-graph query, yaitu graphREL. Dengan berbasis dekomposisi, graphREL bisa menerapkan konsep B-Tree yang biasa hanya dipakai dalam pemodelan relational database..

Kata kunci *: graph database, email, framework relational,* berbasis dekomposisi*, selectivity-aware, graphREL, B-Tree*

PENDAHULUAN

Graph database adalah alat pemodelan dari data yang bersifat sederhana sampai yang kompleks. Pemodelan data kedalam *graph* bisa bermacam-macam. Dari kasus yang diambil (komunikasi data email), terlihat bahwa bentuk data adalah dinamis dan dalam skala yang besar bisa selalu terupdate. Sehingga diperlukan pendekatan sebuah *framework* relasional yang berbasis dekomposisi dan *Selektivitas-Aware* untuk pengolahan *sub-graph query* [Sakr, S. (2009)]. Menurut beliau, GraphREL adalah kerangka murni relasional untuk menyimpan dan query data grafik. GraphREL menurut [1] adalah satu-satunya metode *graph* database yang bisa menerapkan konsep B-Tree. Ini sangat menarik karena B-Tree biasa dipakai dalam pemodelan *relational database*.

Kasus dalam *relational database* tidak dapat menemukan pola untuk mengeksekusi *query* dalam *sub-graph*. Sehingga penulis ingin menerapkan pendekatan graphREL untuk memetakan pola data email kedalam bentuk *graph*, dan untuk mengetahui kecepatan suatu query dalam mengenali pola satu akun mengirim email ke banyak akun, dan untuk mengenali pola satu akun yang hanya menjadi penerima (tidak pernah menjadi pengirim)

METODE PENELITIAN

Memodelkan dataset berupa komunikasi email (*incoming* dan *outgoing* email) kedalam bentuk *graph*. Mengenal pengirim dan penerima pesan dari banyak pengguna dan menciptakan pola-pola dari *graph* kedalam *relational encoding*. Dengan proses *filtering* (*selectivity-aware* dan berbasis dekomposisi), maka akan teridentifikasi hasil berupa *speedup improvement* dari optimisasi *query* relasional [1].

HASIL DAN PEMBAHASAN

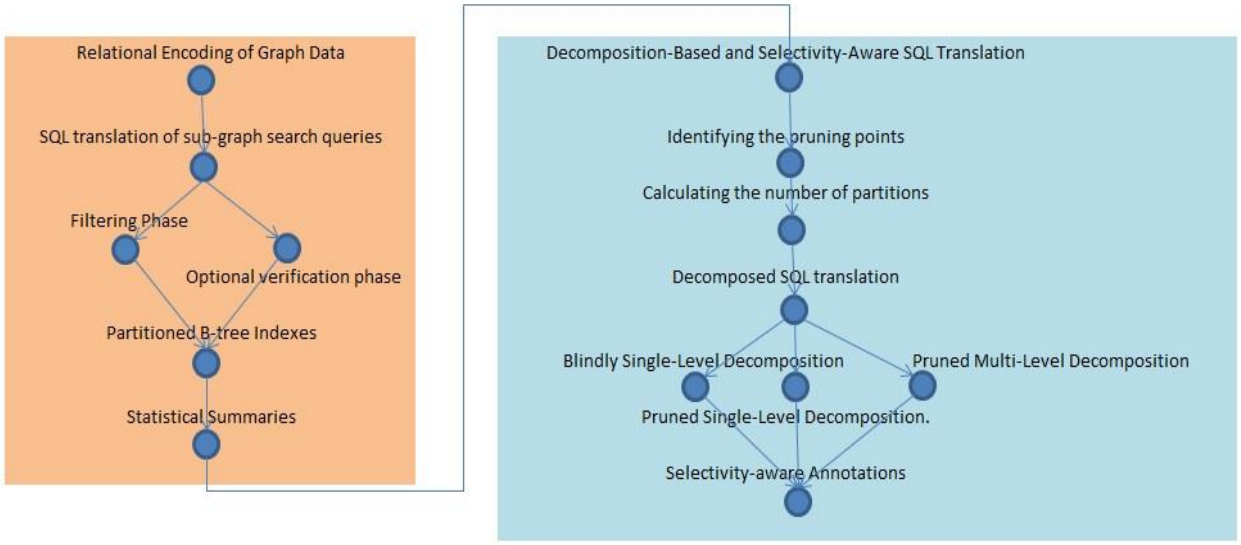
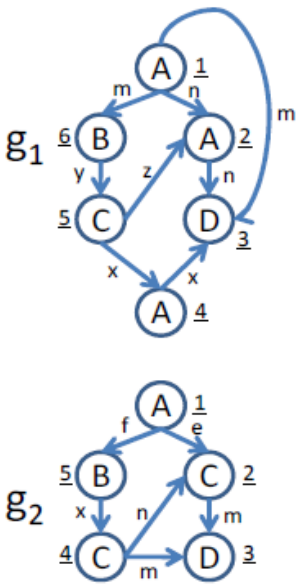


Figure 1 workflow diagram

Gambaran Umum Sistem

1. Pertama-tama adalah memetakan seluruh pola yang ada (*Relational Encoding of Graph Data*) kedalam 2 tabel (*vertices table* dan *edges table*).

   Vertices(graphID; vertexID; vertexLabel)
   Edges(graphID; sVertex; dVertex; edgeLabel)



graphID	vertexID	vLabel
1	1	A
1	2	A
1	3	D
1	4	A
1	5	C
1	6	B
2	1	A
2	2	C
2	3	D
2	4	C
2	5	B

Vertices Table

graphID	sVertex	dVertex	eLabel
1	1	2	n
1	1	3	m
1	2	3	n
1	4	3	x
1	5	4	x
1	6	5	y
1	5	2	z
1	1	6	m
2	1	2	e
2	2	3	m
2	4	3	m
2	4	2	n
2	5	4	x
2	1	5	f

Edges Table

Figure 2-vertices and edges table [1]

2. Pada *SQL translation of sub-graph queries* terdapat 2 fase yaitu *filtering* dan *verification*, semua disajikan kedalam bentuk bahasal SQL untuk memastikan bahwa tiap *vertex* terhadap *graph* adalah berbeda.
3. Tahap partitioned B-tree sedikit berbeda dengan B-tree. Dalam grafik berlabel, umumnya terjadi bahwa *distinct vertices* dan *edges label* jauh lebih sedikit dari jumlah simpul dan tepi masing-masing. *Partitioned B-trees indexes of the high-selectivity attributes* mencapai waktu eksekusi tetap yang tidak lagi tergantung pada ukuran database grafik secara keseluruhan.
4. Secara umum, salah satu teknik yang paling efektif untuk mengoptimalkan waktu eksekusi query SQL adalah memilih eksekusi relasional berdasarkan informasi selektivitas akurat dari predikat *query*. Sehingga pada *statistical summaries* menghasilkan tabel untuk mengetahui *frequency of occurrence* dari tabel yang berbeda (*vertices*, *edges*, dan *edge label connection*).
5. Berdasarkan frekuensi yang ada, penulis bisa mengetahui kemungkinan suatu label merupakan *pruning point* dengan frekuensi yang rendah (NPP). Nilai dari NPP inilah yang akan menentukan jumlah partisi (NOP) yang harus dilakukan *system*. Sehingga dengan partisi yang sudah ditentukan, *system* bisa mendekomposisi *index* dalam graph yang bertujuan untuk mengurangi waktu pencarian. [1].

Pengujian Sistem

Dataset

Dataset yang digunakan dalam pengujian sistem ini adalah data dari SNAP (Stanford Network Analysis Project). Data didapat dari *section large dataset*. Data yang diambil berbentuk *node* dan *edge* yang berjumlah besar (diatas 100 ribu). Data utama dibagi kedalam dua dataset, yaitu euall dan enron.

Untuk EuAll, data email dari lembaga penelitian besar Eropa. Untuk periode dari Oktober 2003 sampai Mei 2005 (18 bulan) kami telah anonim informasi tentang semua email yang masuk dan keluar dari lembaga penelitian. Untuk setiap pesan email yang dikirim atau diterima diketahui waktu, pengirim dan penerima email. Secara keseluruhan data memiliki 3.038.531 email antara 287.755 alamat email yang berbeda.

Tidak jauh berbeda dengan data EuAll, Enron merupakan jaringan komunikasi email mencakup semua komunikasi email dalam dataset sekitar setengah juta email.

Tujuan Pengujian

Tujuan dilakukannya pengujian ini adalah sebagai berikut :

- Mengetahui cara memodelkan data email dalam mencari *query*: pengguna email yang suka mengirim banyak email dan pengguna yang tidak pernah mendapat email (hanya sebagai pengirim).
- Mengetahui hasil implementasi konsep B-Tree terhadap metode GraphREL.
- Membuktikan bahwa pendekatan GraphREL mempunyai keunggulan dalam proses *sub-graph query*.
- Mengetahui hasil evaluasi pendekatan GraphREL terhadap komunikasi data email.

Skenario Pengujian

Pengujian sistem ini dilakukan dengan beberapa percobaan. Percobaan-percobaan ini dilakukan untuk mengetahui pengaruh tiap-tiap parameter pada tiap-tiap dataset terhadap performansi sistem. Berikut ini adalah percobaan-percobaan yang dilakukan :

1. Pengolahan data tanpa tahap filter dan verifikasi (mengabaikan disticnt table)
2. Pengolahan data tanpa partitioned B-Tree
3. Pengolahan data tanpa decomposition dan selectivity aware
4. Pengolahan data dengan tahap filter dan verifikasi (menghilangkan distinct table)
5. Pengolahan data dengan partitioned B-Tree
6. Pengolahan data dengan decomposition dan selectivity aware

Berdasarkan keenam skenario pengujian, diharapkan dapat memberikan perbedaan yang sangat signifikan terhadap setiap hasil pengujian. Dengan begitu dapat dilihat bahwa pendekatan GraphREL adalah tepat untuk kasus tugas akhir ini.

KESIMPULAN

Berdasarkan pengujian yang dilakukan dalam Tugas Akhir ini, dapat disimpulkan bahwa :

a. Graphrel dapat berada pada sistem database relasional dan mengeksploitasi dikenal teknik optimasi matang permintaan serta teknik pengolahan permintaan yang efisien dan terukur nya.
b. Graphrel tidak memiliki biaya waktu yang diperlukan untuk offline atau pra-pengolahan langkah.
c. Graphrel bisa menangani database grafik statis dan dinamis (dengan sering update) sangat baik.

d. Penjelasan selektivitas untuk script evaluasi SQL menyediakan pengoptimalan permintaan relasional dengan kemampuan untuk memilih rencana eksekusi yang paling efisien dan menerapkan efisien pemangkasan bagi anggota database yang non-diperlukan grafik. Semakin banyak proses preprocessing yang dilakukan dalam penanganan *s*sistem prediksi transaksi *online* ini, akan berdampak buruk kepada sistem penanganan forumnya.

DATFAR PUSTAKA

[1] GraphREL: A Decomposition-Based and Selectivity-Aware Relational Framework for Processing Sub-graph Queries. **Sakr, Sherif. 2009.** 2009, Database Systems for Advanced Applications (DASFAA'09).

[2] *A Comparison of a Graph Database and a Relational Database.* **Chad Vicknair, Michael Macias, Zhendong Zhao. 2010.** USA : ACM New York, 2010.

[3] Graph Database Indexing Using Structured Graph Decomposition . **David W. Williams, Jun Huan, Wei Wang. 2007.** Istanbul : IEEE, 2007.

[4] Sorting And Indexing With Partitioned B-Trees. **Graefe, Goetz. 2003.** USA : CIDR, 2003.

[5] Querying RDF Data from a Graph Database Perspective. **Gutierrez, Renzo Angles and Claudio. 2005.** Chile : Springer Berlin Heidelberg, 2005.

[6] Graph Indexing: Tree + Delta >= Graph. **Peixiang Zhao, Jeffrey Xu Yu, and Philip S. Yu. 2007.** Hong Kong : VLDB Endowment, 2007.

[7] Graph Theory. **Ruohonen Keijo** ,2013.