**Review Paper**

**“Mining Frequent Patterns Without Candidate Generation”**

Jika memiliki data yang sangat besar diperlukan biaya yang banyak untuk menangani candidate set dalam jumlah yang cukup besar. Sangat membosankan untuk mengulang pencarian database dan pemeriksaan sebuah set dari kandidat yang besar menggunakan pattern matching.

Langkah pertama yang dilakukan setelah FP-tree dibuat, yang merupakan perpanjangan dari struktur prefix-tree yang menyimpan data penting, informasi kuantitatif tentang frekuensi pattern. Kedua, metode mining fragment growth berdasarkan pattern FP-tree dibuat, yang memulai dari panjang pattern paling sering muncul, menguji hanya dasar pattern tertentu, membangun FP-tree (dengan kondisi tertentu), dan melakukan mining secara rekursif dengan tree tersebut. Ketiga, teknik pencarian yang digunakan dalam penggalian adalah partitioning-based, metode divide-and-conquer.Hal ini menggurangi ukuran dari pola bersyarat base generated secara drastis.

Konstruksi dari FP tree yang tersusun memastikan bahwa subsequent mining dapat dilakukan dengan struktur data yang lebih tersusun. Tetapi tidak berarti secara otomatis memastikan bahwa penggalian ini akan sangat efisien, karena ketika salah satu pattern mungkin masih menghadapi masalah kombinasional candidate generation jika hanya menggunakan FP-tree untuk menghasilkan dan memeriksa semua pola dari candidate.

FP-Growth membangun FP-tree yang sangat padat, sehingga biasanya menjadi lebih kecil daripada database sebenarnya. FP-Growth ini menggunakan pola growth method yang menghindari biaya dengan cara menggabungkan itemset yang sering muncul dalam FP-trees. FP-growth dapat menggunakan pembagian dengan metode divide and conquer yang dapat menggurangi secara drastis ukuran pattern base berikutnya dan FP-trees.