

Comparison of Signature and Inverted Files

Michael J. Nelson
School of Library and Information Science
University of Western Ontario
London, Ontario N6G 1H1

There has recently been a renewed interest in signature method for searching large full text and bibliographic files. The signature method creates signatures which are relatively small strings of bits formed by hashing words or letter combinations and superimposing or concatenating the bit positions set. The signatures are then stored in a separate file and used as a search filter by applying the same algorithm to the query and matching the document signatures to the query signature. Much of the research in this area has analyzed different methods for creating these signatures.

For databases which require a large amount of searching, the method most often chosen is file inversion, which creates indexes which contain information about each word occurrence in a document.

The performance of these methods of searching a database is compared on several different factors, search time, disk space, update time, handling of variable length records, programming complexity and partial word search effectiveness. Two methods of signature searching, word signatures and superimposed coding, and two methods of inverted file organization are compared. The first type of inverted file is a B-tree with one entry for each word occurrence, the second is a dictionary with a tree index and a dense postings file.

After an initial comparison based on theoretical analysis the superimposed coding method and B-tree methods are implemented on a microcomputer to obtain actual performance comparisons on a database of bibliographic citations.

Although the signature methods have an advantage in disk space, update time and programming complexity (except for the bit level operations), the inverted file still has the one big advantage of speed of search which becomes more apparent as the size of database grows. For microcomputer applications and moderate data base sizes, however, the signature search method can still provide acceptable response times of only a few seconds.

To update a signature file, one need only append a new record to the main data base file, compute the signature and append the signature to the signature file. This is contrasted with the relatively complex update of inverted indexes. This makes implementation of signature files for write once read many (WORM) optical especially attractive. Another application for signatures might be differential files of new records added to a database, even for inverted file systems.

Comparaison entre fichiers signés et fichiers Inversés

Michael J. Nelson

School of Library and Information Science
University of Western Ontario
London, ON

On a assisté récemment à un renouveau d'intérêt pour la méthode de repérage par signes dans l'interrogation de grand fichiers bibliographiques ou de texte. La méthode par signes crée d'assez courtes chaînes de bits en effectuant un adressage calculé sur une combinaison de mots ou de lettres, pour ensuite superposer ou enchaîner l'ensemble des positions de bits. Les signes sont ensuite emmagasinés dans un fichier distinct qui sera utilisé comme filtre lors de l'interrogation, alors que la demande sera soumise au même algorithme et les signes produits appariés aux signes du document. Une grande partie de la recherche effectuée dans ce domaine a consisté à analyser les méthodes utilisables pour créer ces signes.

La méthode d'indexation la plus souvent choisie dans le cas des bases de données fréquemment interrogées est celle de l'inversion des fichiers. Cette méthode consiste à créer des index contenant de l'information sur la présence de chacun des mots d'un document.

La performance de ces deux méthodes de recherches dans des bases de données fait l'objet de comparaisons basées sur plusieurs facteurs: la durée de la recherche, l'espace disques, le délai de mise à jour, la prise en charge des articles de longueur variable, la complexité de programmation et l'efficacité de la recherche par caractères de remplacement. On compare deux méthodes de recherche par signes, signes de mot et codage superposé, et deux modes d'organisation de fichiers inversés. Le premier de ces modes consiste en un arbre binaire qui contient un élément pour chaque occurrence d'un terme, et le second consiste en un dictionnaire comprenant un index arborescent et un fichier à inscriptions denses.

Suite à une comparaison préliminaire effectuée sur la foi d'une analyse théorique, les méthodes de codage superposé et d'arbre binaire ont été mises en oeuvre sur micro-ordinateur, ceci pour en comparer l'efficacité dans le cas d'une base de données bibliographique.

Bien que la méthode par signes possède l'avantage en ce qui a trait à l'utilisation d'espace disques, au délai de mise à jour et à la complexité de programmation (sauf pour les opérations au niveau du bit), le fichier inversé quant à lui possède toujours l'avantage majeur d'une plus grande vitesse de recherche. Cet avantage devient de plus en plus apparent à mesure que la taille de la base de données s'accroît. Au niveau des applications sur micro-ordinateur cependant, la méthode de recherche par signes permet d'offrir un temps de réponse, acceptable, de quelques secondes, en autant que la taille de la base de données demeure modeste.

Il suffit, pour mettre à jour un fichier indexé par signes, d'ajouter la nouvelle notice à la fin de la base de données principale, de calculer le signe et de l'ajouter à la fin du fichier de signes. Ceci se compare avantageusement à la complexité de la mise à jour des fichiers inversés et rend l'utilisation de fichiers de signes pour les disques optiques à écriture unique et lecture multiple (WORM) particulièrement attrayante. Également attrayante serait leur utilisation pour créer des fichiers différentiels de notices ajoutées à une base de données, et ce même pour des systèmes faisant appel à des fichiers inversés.