

ハイブリッド型文書検索システムの試作と性能評価

牛嶋 一智[†] 今一 修[†] 安田 知弘[†] 岩山 真[†]

† 株式会社 日立製作所 中央研究所 〒185-8601 東京都国分寺市東恋ヶ窪 1-280

E-mail: [†]{kazutomo.ushijima.mv, osamu.imaichi.xc, tomohiro.yasuda.bm, makoto.iwayama.nw}@hitachi.com

あらまし 企業内に蓄積された大量業務情報の有効活用を目的として、大規模高度検索技術へのニーズが高まっている。検索処理の大規模化および高度化に対しては、高速な絞込み検索が得意なインデックス型検索と高度な検索が得意なスキャン型検索を組合せたハイブリッド型のアプローチが有効である。本稿では、ハイブリッド型検索の実応用での性能特性確認のため、ハイブリッド型を採用した文書検索プロトタイプを開発し、両検索方式の性能特性に応じて適切な処理分担を行う検索式変換法の検討を行った。その結果、実検索履歴に基づく特許検索ベンチマークにおいて、ハイブリッド型検索の有効性を確認するとともに、性能特性に応じた検索式変換により最悪検索時間を3割強、平均検索時間を2割強さらに削減できる見通しを得た。

キーワード ハイブリッド検索、インデックス検索、スキャン検索、検索式変換、文書検索

Design and Evaluation of Hybrid Document Retrieval Prototype System

Kazutomo USHIJIMA[†] Osamu IMAICHI[‡] Tomohiro YASUDA[‡] and Makoto IWAYAMA[‡]

† Hitachi, Ltd., Central Research Laboratory 1-280, Higashi-koigakubo Kokubunji-shi, Tokyo 185-8601 Japan

E-mail: [†]{kazutomo.ushijima.mv, osamu.imaichi.xc, tomohiro.yasuda.bm, makoto.iwayama.nw}@hitachi.com

Abstract Large-scale advanced search techniques are getting important to realize efficient utilization of growing enterprise data. In general, index-type search methods are suitable for large-scale search and scan-type search methods are suitable for advanced search. Combining these types of the search methods, we can construct hybrid-type search methods effective for both types of search. In this article, we describe a prototype system which presents a practical performance profile of a hybrid-type text search method and also propose a search formula translation technique which modifies the assignment of search processing between the search methods according to their performance profiles. As a result, we confirmed the effectiveness of the hybrid-type search method and also confirmed an over 30 percent reduction in the worst search time and an over 20 percent reduction in the average search time using the search formula translation technique.

Keyword Hybrid Search, Index Search, Scan Search, Search Formula Translation, Document Retrieval

1. はじめに

近年、企業内への急速なIT技術の浸透により、様々な業種及び業務の企業活動において、電子的に蓄積される業務データのサイズが爆発的に増加しつつある。また社内に蓄積された大量の業務データを単に保有するだけでなく、関連業務へと活用範囲を拡大するために共通検索基盤を整備し、社内業務効率や顧客サービスレベルの向上に繋げることが、業界における競争力確保に不可欠な要因となりつつある。例えば、ECサイトにおいて、ユーザ購入履歴に基づき好評な商品を優先して拡販を行ったり、コールセンタにおいて、過去に着信した顧客対応記録に基づき、新規顧客のタイプを判断し、適切な受け答えを実施したりという多くの業種において一般的になりつつある。

企業内に蓄積される業務データは、ディスク装置の大容量化、ICカードやRFIDなどの情報収集手段の普及等を受け、急速にそのサイズ拡大を進める[1]とともに、収集データに付随する文脈情報や意図にまで踏み込んだ高度な検索に対するニーズも高まりつつある[2]。特に、企業業務データを検索対象とする企業向け検索システムでは、多種多様雑多な情報を対象とするWeb検索と異なり、データ形式やデータ内容に一定の共通部分が存在し、比較的均質なデータの存在を前提とした検索処理を行うことが出来る。すなわち、検索

対象データに対して一定の業務モデルを前提として、状況や文脈に関する情報を抽出し、それらを活用した検索を行なうなど高度な検索の実施が可能である。

我々は、企業情報検索における「検索対象規模拡大」及び「検索内容高度化」という2つの課題を踏まえ、企業情報向け共通検索基盤の検討を行った。共通基盤のアーキテクチャ検討としては、適用先の検索要件に応じて複数の検索方式を組合せができるハイブリッド型検索方式の検討を行った。特に同様の検索機能を有しながら性能特性の異なる検索手法を組み合わせることで、上記2つの課題に対応する新たな性能特性を有する検索手法を提供することが可能となる。

本稿では、ハイブリッド型検索の文書検索分野での有効性確認を目的として、既存の高速インデックス型文書検索エンジンをベースに開発したハイブリッド型文書検索プロトタイプについて述べるとともに、実検索履歴に基づく特許検索ベンチマークを用いて行った性能評価結果を示す。その結果、ハイブリッド検索の採用により、大幅なメモリ使用量増加を伴わずに、スキャン単独方式と比較して最悪検索時間を約6割に削減、平均検索時間を2%以下に減少させることを確認した。加えて、それぞれの検索エンジンの性能特性に応じて検索式を変換し、両検索エンジンでの検索内容を効率的に分配する技術の検討を行い、単純な連携方式と比

較して最悪検索時間を3割強、平均検索時間を2割強さらに削減できる見通しを得た。

2. 企業情報検索における需要動向

本節では、企業情報検索における需要動向と企業情報検索向けの共通検索基盤システムの検討方針について述べる。

2.1. 検索対象範囲及び規模の拡大

ストレージ装置の大容量化、ICカードやRFIDなどの情報取得手段の普及、さらには業務プロセスの可視化の必要性の高まりなどを受けて、企業内において検索対象となる業務情報の拡大が進んでいる。例えば、報告書類や設計文書など従来から企業内に蓄積されていた情報に対する検索から、メールやワークフロー文書、さらには法令順守対応のための入退室記録や業務操作ログなど、業務プロセスの進行に伴い発生する大量の履歴情報についてもごく一般的に検索対象とされるに至った。

2.2. 検索処理の高度化

一方で、企業内での業務効率向上やより高度な顧客サービス提供を目的として、事実抽出を行う検索や文脈を考慮した検索などデータの内容に踏み込んだ高度な検索に対する需要が拡大している。例えば、製薬会社向けに医療文書において病名と化合物名を区別した検索を行なえるようにして、利用者の目的に即した検索結果を迅速に取得できるようにしたり、オンラインショップにおける顧客のクリックストリーム履歴から顧客の購買意図などを推測し、販売商品の趣向と顧客ニーズの合致までを考慮しながら推薦商品の選択を行なう高度な商品紹介サービスを提供したりといったことが可能となる。

2.3. 情報検索基盤に向けた検討

我々は、企業情報検索における以上の動向分析を踏まえ、スケーラブルで機能拡張が可能な企業向け情報検索基盤システムの実現に取り組むこととした。また、基盤システムの基本アーキテクチャ検討の一環として、インデックス型検索とスキャン型検索を組合せたハイブリッド型文書検索の検討にまず取り組んだ。ハイブリッド型文書検索では、それぞれ一定の文書検索機能を有する検索エンジンを組合せて検索を行うため、検索内容の特性に応じて両者で柔軟に検索処理を分担することができる。

3. 各検索手法の性能特性

本節では、文書検索において代表的なインデックス型及びスキャン型の各検索技法の概要とその性能特性について述べる。

3.1. インデックス型検索

大量文書群に対する高速検索を実現する代表的なインデックス型検索技法としては、転置リスト法[3]を挙げることが出来る。転置リスト法では、事前に指定されたトークン(検索用文字列)集合について、各トークンを含む全文書IDをまとめた文書IDリストをインデックスとして作成する。検索時には検索式に出現する各トークンについて、対応する文書IDリストを参照し、

指定された論理式に従った判定評価を行うことで検索条件を満たす文書ID集合の絞込みを行う。検索に利用するトークン集合としては、形態素(事前に辞書登録された文字列)あるいはN-gram(検索対象テキストからN文字毎に機械的に切り出した文字列)の選択肢がある。新語や専門用語に対する検索漏れが好ましくない企業内検索においては、N-gramが広く利用されている。

転置リストを用いた検索では、論理式評価の際に参照される文書IDリスト長の合計(総DF値: Total Document Frequency)に比例する検索時間が必要である。一般に検索対象文書に頻繁に出現する検索語(一般語)に含まれるトークンのDF値は大きくなるため、一般語を多く含む検索式や多数の検索語を含む複雑な検索式の場合、インデックス型での検索時間が大きく増大するという性能特性がある。また転置リスト法において、N-gramの語としての出現連続性や近傍検索をインデックス上で行うために、文書IDリストにおいてトークン出現位置情報を保持を行うと、インデックスサイズは元文書の数倍に肥大する。逆に出現位置情報を保持せず、インデックス側ではトークンの出現判定のみを行い、さらに文書IDリストの圧縮などすることで、インデックスサイズを元文書の半分以下とし、オンメモリでの高速検索を行うことも可能となる。

3.2. スキャン型検索

文書データに対するスキャン型検索では、指定された検索式をオートマトンなどのスキャンプログラムに変換し、検索対象データを順次スキャンすることで検索対象データが検索条件を満たすかを判定する。一般にスキャン型の検索は、データサイズに比例した検索時間が掛かるが、検索対象データをその場でフルスキャンするために、前後の文脈を考慮した検索等の柔軟な処理が容易に実現できる。ただし、以下ではスキャン型検索のうち最も単純な文字列一致検索に限って検討を行なう。

文字列一致検索のためのスキャン方式に関しては、オートマトン等の構成の違いからAC(Aho-Corasick)法[4], CW(Commentz-Walter)法[5], WM(Wu-Manber)法[6]等が広く知られており、それぞれの検索方式は、検索式中の最短検索語長や異なり文字種類数などに応じてそれぞれ性能特性が異なる。例えばCW法では、検索式に含まれる検索語長に応じてスキャン対象文書を読み飛ばすため、検索式に短い検索語が含まれた場合、検索性能が大きく劣化する。

4. ハイブリッド型検索プロトタイプの開発

インデックス検索では、インデックスを用いて大量データを高速に絞り込むことが出来るのに対して、高度な検索を行なうために付加的な情報の保持を行なうとインデックスサイズが肥大化する。一方、スキャン検索では、検索対象データをフルスキャンすることから、複雑な解析処理を必要とする高度な検索を実行可能であるのに対して、検索対象データサイズに比例する検索所要時間が必要となる。このとき両者の検索技法を組合せることで、大規模データに対しても高度な検索処理を高速に実行可能な検索エンジンの構成が可能となる。我々は、文書検索応用におけるハイブリッド検索の性能特性検証を行なうために、オンメモリでの高速検索が可能なインデックス型検索エンジンMANTA

(Multi-purpose Analysis for Transposable Association)[7]に対してスキャン型エンジンを組合せたハイブリッド型の検索プロトタイプの開発を行った。

4.1. 性能特性に応じた検索処理の分担

プロトタイプ開発のベースとなる MANTA は、検索対象とその特徴量の対応付けを、オンメモリに保持した圧縮インデックスを用いて両方向に高速に検索することが可能な汎用連想検索エンジンである。MANTA の検索対象を文書、その特徴量として文字 2gram を利用した場合、投入された検索式に含まれる検索語を全て文字 2gram に変換し、検索を行なうことにより、検索ノイズは含まれるが元の検索式を満たす可能性のある文書集合を漏れなく検索することが出来る。

また、本プロトタイプのスキャン検索モジュールとしては、新たに作成したスキャン検索エンジンを採用した。本スキャン検索エンジンは、投入されたスキャン検索条件に従って文書データ文字列に指定検索語が含まれるかどうかを判定するオートマトン構成し、検索語の出現に関する論理条件判定を行った後、タグ指定による出現位置判定および検索語の出現位置の近傍判定を行う。スキャン方式としては、一般的なパラメータ設定にて最も性能特性のバランスが良い CW 法を採用した。

両検索エンジンの組合せにおいては、インデックス側では、トークン出現位置情報を保持せずインデックスをオンメモリに留めたまま、トークンの出現判定のみを行い、スキャン側で最終的な検索語としての判定を行うという機能面での分担を行った。さらに加えて本プロトタイプでは、両検索エンジンの性能特性を考慮し、以下のような内容面での処理分担調整も行うことによって、全体の検索時間の短縮を図ることとした。

1) 総 DF 値に閾値を設け、それを超える検索トークンの検索はインデックス検索側では行わない。

2) スキャン検索側は、語長の短い検索語に関する判定はインデックス検索側の判定結果を参照し、その検索語に対する条件評価は行わない。

4.2. 検索式変換

ハイブリッド型検索システムにおける検索エンジン間での処理分担の変更は、具体的にはそれぞれの検索エンジンに投入される検索式を適切に修正することで実現される。

4.2.1. インデックス検索用検索式の変換

文書 ID リストを用いたインデックス検索では、検索式に DF 値の大きいトークンが多く含まれた場合に検索時間が増大する。特に、ほとんど全ての文書に出現するようなトークンは、検索式の評価に利用した場合に文書 ID 集合をあまり絞り込むことが出来ず、かえって検索時間を増大させてしまう可能性がある。ハイブリッド検索では、インデックス検索の後段にスキャン検索が控えているため、一部のトークンに関する検索条件判定を省略しても、スキャン検索側でチェックすることで、全体としての検索式評価の等価性を保持することが出来る。ただし、あまりインデックス側で検索条件を緩めすぎると、スキャン検索側に引き渡される検索対象文書数が増大し、スキャン検索側の検索時間が増大することになる。任意の論理式と文書 ID リス

トが与えられた場合に、全体検索時間を最適とするトークンの組合せを求めるることは困難であるため、本プロトタイプではインデックス側での最悪検索時間の目安となるパラメータとして総 DF 値閾値を導入し、以下の近似的手法によりインデックス検索側で検索を行うトークンを決定することにした。

検索式変換手順：

- 1) 検索式に含まれる全てのトークン(重複を除く)について、DF 値を求め昇べきにソートする。
- 2) DF 値の小さいトークンから順に累計を計算し、事前に決定した総 DF 値閾値を超えた後は、その後のトークン(例:「特徴」)については、検索式におけるそのトークンの出現を全て 'T' (恒真を表す特殊記号) に変換する。

例：((静止×止画+動画)×(特徴×微素))×検索
→ ((静止×止画+動画)×(T×微素))×検索

- 3) ただし上記の変換において、検索判定を省略するトークンは、後続のスキャン検索において必ず条件判定が行われることが前提である。そのため、後述のスキャン検索用検索式の変換において、スキャン側で条件判定が省略されるトークンについては、「T」への変換を行なわない。
- 4) ただし、変換後の検索式が検索対象文書に係わらず常に真となる恒真式となってしまった場合(例えば('T'×'止画')+'T' といった形)は、インデックス側での絞り込みがまったく行われなくなってしまうので、元の検索式を用いる。

4.2.2. スキャン検索用検索式の変換

CW 法に基づくスキャン検索では、検索式に 2 文字以下の検索語が含まれる場合、全体のスキャン検索時間が大きく増大する。プロトタイプのインデックス検索エンジンでは、3 文字以上の検索語については 2gram に分割して検索するために、それぞれの 2gram が連続して出現するかどうかをスキャン検索において確認することで初めて、検索語の出現の有無が判定される。一方、検索語が 2 文字以下の場合は、インデックス検索側で検索語の出現の有無について完全に判定が行えるため、該当する検索語に関する出現判定をインデックス側のみで行い、スキャン検索用検索式から省略することで、スキャン検索時間の短縮を図ることができる。

例：(静止画+動画)×特徴素×検索 → (静止画+?)×特徴素×

4.2.3. 変換後検索式を用いた検索の実行

ハイブリッド型検索エンジンでの検索式の変換と変換後の検索式を用いた検索処理の例を図 1 に示す。以下のような検索手順により、ハイブリッド検索において、最悪検索時間と平均検索時間を大幅に改善させた検索処理を実行することが可能となる。

検索実行手順：

- 1) まずユーザから投入される検索式をインデックス検索用検索式とスキャン検索用検索式に分ける。
- 2) インデックス検索用検索式については、検索式を構成する検索語を 2gram に分割し、インデックス検索エンジン用の検索式を用意する。
- 3) 分割後の 2gram の集合を、(A) 検索語が分割され

- て得られた 2gram の集合と(B)検索語が分割されずに得られた 2gram の集合とに分類する。
- 4) (A) 検索語が分割されて得られた 2gram の集合については、それぞれの 2gram の DF 値に従つて昇べきにソートする。累計 DF 値が総 DF 値閾値を越えた以降の 2gram(例:‘特徴’)については、インデックス側での条件判定を省略することとし、インデックス検索用検索式において、該当する全ての 2gram を全件ヒットに相当する恒真値‘T’に変換する。
 - 5) (B) 検索語が分割されずに得られた 2gram の集合については、後続のスキャンにおいてインデックス側の判定結果を参照できるよう、判定結果を一時格納するためのメモリ領域を確保する。
 - 6) 2)-5)の作業の結果得られたインデックス検索用検索式を用いてインデックス検索を行ない、文書 ID 集合の絞込みを行なう。インデックス検索では、まずインデックス検索用検索式に含まれる全ての 2gram について、対応する文書 ID リストを取得する。統いて、それぞれの文書 ID リストを最も若い文書 ID 順に参照し、インデックス検索用検索式の各 2gram について、注目する文書番号がその 2gram の文書 ID リスト中に含まれる場合はその 2gram を‘T’に、含まれない場合は‘F’に変換する。このとき変換後の論理式全体の評価結果が‘T’になった場合に、その文書 ID がインデックス検索でヒットしたとする。
 - 7) 一方、スキャン検索用検索式に関して、2 文字以下の検索語については、スキャンを行なわず、インデックス検索側の判定結果を参照することを示すため、その検索語を不定真偽値をあらわす‘?’へと変換を行なう。
 - 8) 6)のインデックス検索の結果得られた個々の文書 ID について、対応する文書本体を取得し、7)で得られたスキャン検索用検索式を用いてスキャン検索を行なう。このとき、不定真偽値以外の検索語の出現の有無について CW 法に基づくオートマトンにて判定を行なった後、不定真偽値の値としてインデックス側での判定結果を参照して、最終的なヒット判定を行なう。

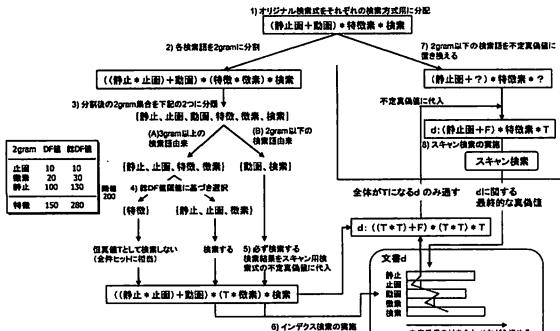


図 1: ハイブリッド検索における検索手順の流れ

5. プロトタイプ設計

本節では、今回開発したプロトタイプの設計概要について述べる。

5.1. 全体構成

本プロトタイプは、図 2 に示すように全体の実行制御と投入された検索式をそれぞれの検索モジュール向けて変換及び配布を行う全体制御モジュール、配布された検索式に従って文書 ID レベルで検索対象データの絞込みを行うインデックス検索モジュール、検索対象データ本体の登録管理を行う検索対象データ管理モジュール、同じく配布された検索式に従って検索対象データ本体を参照してスキャン判定を行うスキャン検索モジュールからなる。

インデックス検索モジュールから検索対象データ管理モジュールへの絞り込み済み文書 ID 集合の受け渡し、および検索対象データ管理モジュールからスキャン検索モジュールへの文書格納先情報の受け渡しは、それぞれ専用の FIFO キューを用いて行い、上記各モジュール間の同期もこの FIFO キューを介して行う。以下、それぞれのモジュールの概要について述べる。

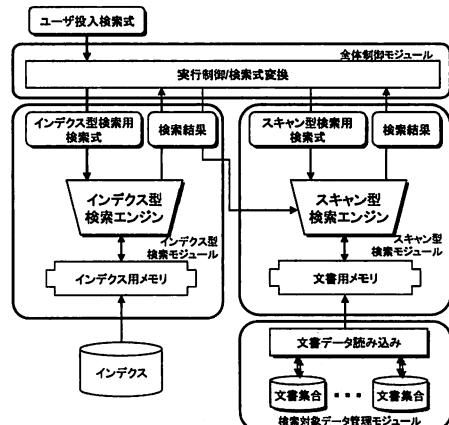


図 2: プロトタイプモジュール概要図

5.2. 全体制御モジュール

全体制御モジュールは、実行制御部および検索式変換部から構成される。実行制御部は、インデックス検索モジュール、検索対象データ管理モジュール、スキャン検索モジュールの 3 つのモジュールの実行制御を行う。各モジュールには、それぞれ独立にスレッドが割り当てられ、検索要求毎に各検索モジュール間で同期をとりながら動作を行う。検索式変換部では、ユーザーから投入された検索式をそれぞれの検索モジュールに適した検索式に変換して配布する。

5.3. インデックス検索モジュール

インデックス検索モジュールでは、全体制御モジュールから受け取ったインデックス検索用検索式にしたがって、オンメモリに保持したインデックスを参照し、検索式を満たす可能性がある検索文書の ID 集合を絞り込む。絞り込まれた文書 ID 集合は、文書 ID キューを介して検索対象データ管理モジュールに受け渡される。

5.4. 検索対象データ管理モジュール

検索対象データ管理モジュールでは、インデックス検索モジュールで絞り込まれた各文書 ID に対応する文書本体のメモリ上の格納位置情報(開始アドレスとサ

イズ)を文書格納先情報として、文書情報キーを介して、スキャン検索エンジンに受け渡す。文書 ID と文書本体の対応付けは、文書登録時に作成されるインデックスを介して取得される。

5.5. スキャン検索モジュール

スキャン検索モジュールでは、検索対象データ管理モジュールから受け取った文書格納先情報に基づき文書本單の取得を行ない、文書本体に対してスキャン検索用の検索式を満たすかどうかの判定を行う。インデックス検索用の検索式は、検索対象文書数を絞り込むための緩い検索条件であるが、スキャン検索用の検索式と組合せることで投入された検索式と等価な検索処理が実行できることになる。

6. 性能評価

6.1. 性能評価条件

今回の性能評価に用いた検索対象特許データは、公開済み特許明細書データから約 3 万件を目処に抽出した約 800MB を利用した。実際に検索システムを構築する場合、数百台のサーバ機を並べたクラスタ構成が採用されることを想定し、今回の性能測定では基礎データ取得のため、サーバ機一台に注目した構成での性能評価を行なっている。また検索式は、社内特許検索システムの検索ログから、全文検索を行っている検索式 9,488 件を抽出し、これらを用いてプロトタイプの性能特性評価を行なった(インハウスシステムでの実績)。また本特許文書を対象として作成した MANTA の圧縮インデックスのサイズは、元文書の 18% であった。

6.2. 性能評価結果

ハイブリッド方式：検索式を変換しない場合

インデックス型検索エンジンとスキャン型検索エンジンを組合せたハイブリッド方式において、検索式の変換をまったく行わない場合の性能測定結果を図 3 に示す。本グラフにおいて、横軸は検索結果件数、縦軸は検索所要時間(相対値)を表す。それぞれの検索式を使って検索対象データの検索を行なった場合の検索所要時間を全体検索所要時間(SEARCH)とし、その内訳となるインデックス検索所要時間(INDEX)及びスキャン検索所要時間(SCAN)と共に示す。このとき、グラフに示す検索所要時間には、検索式の変換時間、及び検索結果の出力時間は含まれていない。以下では、本ケースでの最悪検索時間を 1.0 として検索所要時間を示す。

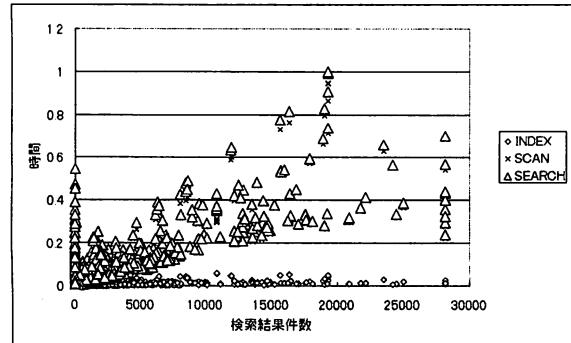


図 3: 検索式を変換しない場合

グラフの Y 軸近辺においては、インデックス検索時間の増大、また検索結果件数 2 万件近辺においては、スキャン検索時間の増大がそれぞれ全体検索時間の悪化を引き起こしていることがわかる。また最悪検索時間を 1.0 とした場合の平均検索時間は 0.0213 であった。

ハイブリッド方式：インデックス用検索式のみ変換

次に総 DF 値閾値を導入し、インデックス検索側で DF 値大のトーケンを省略して検索を行なった場合の性能評価結果を図 4 に示す。Y 軸近辺の測定データにおいて、インデックス検索時間が減少し、全体検索時間が大幅に改善していることがわかる。このとき、最悪検索時間は 0.989、平均検索時間は 0.0203 であった。

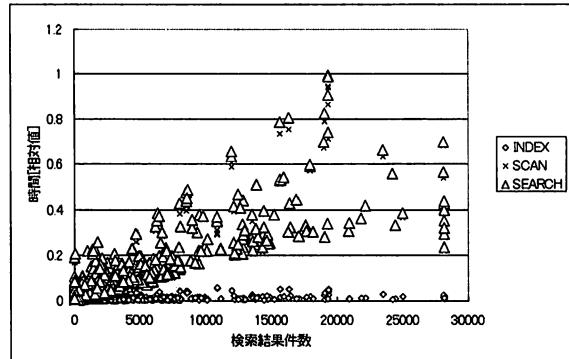


図 4: インデックス用検索式のみ変換

ハイブリッド方式：両検索側で検索式を変換

さらにスキャン検索用検索式についても検索式の変換を行い、2 文字以下の検索語の判定処理をインデックス側で行なうようにした場合の性能特性を図 5 に示す(プロトタイプを等価動作させて得られた実測値)。

グラフの検索結果件数 2 万件近辺において、スキャン検索時間が短縮したことにより、全体検索時間が改善していることがわかる。最悪検索時間及び平均検索時間も、それぞれ 0.657, 0.0160 と大きく改善した。

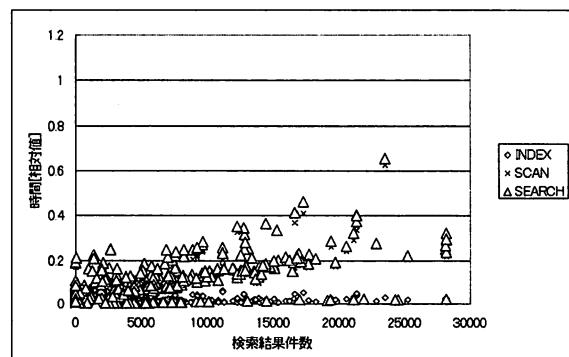


図 5: 両検索側で検索式を変換した場合

性能評価結果まとめ

スキャン型検索エンジンのみを用いて同等の検索を行なった場合、最悪検索時間は 1.63、平均検索時間は 1.37 との測定結果であった。すなわち、ハイブリッド

方式は、スキャン検索単独方式と比較した場合、最悪検索時間を約6割に削減、平均検索時間を2%以下に減少させることができ、さらに検索エンジンの性能特性に応じて適切な処理分担を行なうことで、さらに最悪検索時間を3割強削減、平均検索時間を2割強削減できる見通しを得ることができた。(図6,7)

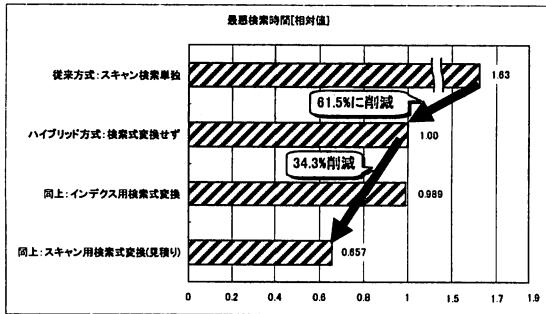


図6: 最悪検索時間比較

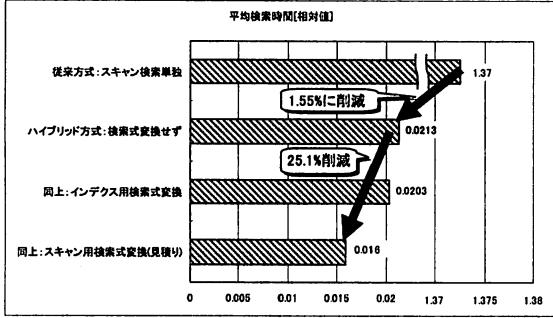


図7: 平均検索時間比較

7. 関連研究

本稿における関連研究としては、文字成分表あるいはシグネチャファイルを検索対象の絞り込みに用いる階層プリサーチ方式[8]を挙げることが出来る。階層型プリサーチは、文字成分表等を用いて検索対象を絞り込んだ後、対応する凝縮テキストあるいはテキスト全体の検索を行う。このとき文字成分表に登録する文字列を工夫することで、前段絞り込み効率を改善することができる。本稿では、前段のインデックス検索エンジンが後段のスキャン検索エンジンに対して絞込みを行った後に検索結果を受け渡すだけでなく、後段の検索エンジンが前段の検索判定結果の一部を参照するという両方向の連携を行うことにより、さらなる性能向上を実現している。

また同じく検索対象のオンメモリ化により文書検索の高速化を狙った関連研究として、オンメモリスキャナにより安定的な検索所要時間の提供を実現するスキャン単独型検索エンジン[9]を挙げることができる。スキャン単独型の検索方式では、検索対象文書をオンメモリに保持しフルスキャナを行うため、インデックスの作成を必要としない。従来のインデックス単独型の検索システムと比較した場合、スキャン単独型の検索システムは、平均検索時間では劣るもの、大容量メモリを活用し最悪検索時間を対話的な検索が可能な範囲

に収めることで、一定のユーザニーズを獲得している。

今回開発を行ったハイブリッド型検索プロトタイプでは、オンメモリ上のインデックス絞込み技術とスキャナ技術を連携させ、両者の利点を兼ね備えた検索処理を効率良く行なうことが可能であり、同等のメモリ使用量において、最悪検索時間、平均検索時間ともに単独型のアプローチを大きく上回ることができた。今後、検索対象の大規模化と検索内容の高度化が予想される企業検索応用においては、インデックスサイズの肥大を抑えながら、高速な検索処理を実現することができるハイブリッド型のアプローチが有望である。

8.まとめ

我々は、ハイブリッド型アプローチに基づく文書検索プロトタイプを開発し、実検索履歴に基づく特許検索ベンチマークを用いて性能評価を行なった。その結果、スキャン単独方式と比較して、メモリ使用量を大幅に増加させることなく、最悪検索時間を約6割に削減、平均検索時間を2%以下に減少させることができた。また、検索エンジンの性能特性に応じて両検索エンジンで検索する処理内容を分配する検索式変換技術を用いて、さらに最悪検索時間を3割強、平均検索時間を2割強削減できる見通しを得た。

今後は、ハイブリッド型検索のアプローチを、文脈や意味を考慮した検索に代表される、より高度な検索手法に対して適用し、その適用範囲の拡大を検討していく。

文献

- [1] School of Information Management and Systems, University of California at Berkeley, "How Much Information? 2003", (<http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/>).
- [2] 山名早人, 「検索エンジンから分析エンジンへ」, 人工知能学会誌, Vol.20, No.4, pp.471-478, 2005.
- [3] 北研二, 津田和彦, 獅子堀正幹, 「情報検索アルゴリズム」, 共立出版, 2002年出版.
- [4] Aho, A. V. and Corasick, M. J., "Efficient string matching: an aid to bibliographic search", CACM, 18(6), pp.333-340, 1975.
- [5] Commentz-Walter, B., "A string matching algorithm fast on the average", In Proceedings of the 6th International Colloquium on Automata, Languages and Programming. LNCS71, pp.118-132, 1979.
- [6] Wu, S. and Manber, U., "A fast algorithm for multi-pattern searching", Report TR-94-17, Department of Computer Science, University of Arizona, 1994.
- [7] 安田知弘, 今一修, 岩山真, 丹羽芳樹, 「連想検索エンジンのスケーラビリティおよび障害耐性の向上」, 情報処理学会第69回全国大会, 3D-1 1-383-384, 2007.3.
- [8] 島山敦, 浅川悟志, 加藤寛次, 「ソフトウェアによるテキストサーチマシンの実現」, 情報処理学会情報学基礎研究報告, Vol.92, No.32, 25-2, pp.19-25, 1992.5.
- [9] 日経BP企画・編著: 革新的XML型データベースエンジン「Shunsaku」, 2004.