4N－8

大規模テキスト並列検索エンジン RetrievalExpress * (2) 構造化テキスト検索方式

赤峯享　福島俊一　米山千美　清沢治彦　田中俊行 †
NECヒューマンメディア研究所　NEC情報システムズ ‡
e-mail: akamine@hml.cl.nec.co.jp

1　はじめに

近年、SGMLやMTMLに代表される、一件のテキスト中に「題」や「1章」等の構成要素をもつ構造化テキストが、インターネット／イントラネット上で頻繁に用いられるようになっている。それにつれて、特定の構成要素のみを検索対象として指定する検索が、精度の高い検索を実現する上で、重要な機能になっている[1]。

筆者らは、検索洩れ・検索ノイズのない高速全文検索方式としてフレキシブル文字列インバージョン法を提案し、さらにインデックスを分割し並列検索を行うことでスケーラビリティの改善を行ってきた[2, 3]。本稿では、フレキシブル文字列インバージョン法をベースとしたゾーン検索機能を中心に全文検索エンジンRetrievalExpressの構造化テキスト検索方式を述べ、その検索性能の評価結果を報告する。

2　構造化テキスト検索機能

構造化テキストに関する検索は、全文検索対象になるテキスト本文の特定の構成要素(「題」、「1章」等)に対する検索(ゾーン検索)と、事項検索の対象となる書誌事項(「日付」、「テキスト番号」等)に対する検索(フィールド検索)に分けられる。以下、RetrievalExpressのゾーン検索とフィールド検索の実現方法について述べる。

2.1　ゾーン検索

ゾーン検索は、テキストの特定の構成要素中に出現する文字列に対する全文検索を行う。例えば、ゾーン「題」に「検索」という文字列を含むテキストを検索する場合、「題＝“検索”」というような検索条件式で検索する。

ゾーン検索機能の実現は、フレキシブル文字列インバージョン法をベースとした。フレキシブル文字列インバージョン法は、キー文字列を格納するキー情報部と、テキストＩＤとテキスト内位置を格納する位置情報部からなるインデックスを利用する全文検索方式であり、以下の特徴を備えることでディスクへのアクセスの量を減らし、高速な全文検索を実現している[2]。

(1) キー文字列長の字種別可変化。
(2) キー文字列への縮退文脈の付与。
(3) 高頻度文字列用の副インバーテッドファイルの併用。
(4) 位置情報データの圧縮。

テキストＩＤとテキスト内位置をもつタイプのインデックスを用いてゾーン検索を実現する方法として、以下の3種類が考えられる。

(A) ゾーンに関する情報をキー情報部にもつ。
(B) ゾーンに関する情報を位置情報部にもつ。
(C) 全文インデックスとは別のゾーン範囲情報のテーブルをもつ。

(A)のタイプでは、インデックスのキー情報としてキー文字列とゾーン名を併せて持ち、検索時にキーワードとゾーン名をキーとしてインデックスを検索することでゾーン検索を実現する。(B)のタイプでは、位置情報部にテキスト識別とテキスト内位置以外にゾーン名の情報も併せてもち、検索時にキー文字列に対応する位置情報からゾーン名が一致する部分だけを選択することでゾーン検索を実現する。(C)のタイプでは、各テキスト毎に各ゾーンの開始位置と終了位置を記憶しておき、検索時に該当した各テキスト毎にキーワードがゾーン内に出現することをチェックすることでゾーン検索を実現する。

しかしながら、(A)のタイプのインデックスを利用した場合、ゾーンの数に比例してキーが増加してしまい、キー情報部のインデックス容量が増加して、キーを検索する速度の低下という問題を生じる。(B)のタイプでは、位置情報部のインデックス容量が増加してしまい、位置情報部の読み出し量の増加によって検索速度が低下してしまう。(C)のタイプでは、各テキスト毎の各ゾーンの開始位置と終了位置を記憶して、該当テキストのそれぞれに対してゾーンの開始位置と終了位置の検索を行う必要があるため、この処理がオーバーヘッドになって、検索速度が低下してしまう。

これらの問題点を回避するため、RetrievalExpressでは、ゾーンの位置情報を全てのテキストで共通になるような形に一旦変更し、その位置情報に対してインデックスを作成する方法をとった。この方法は、(B)のタイプと比較した場合、インデックスのテキスト内位置自体にゾーンの情報を持たせることで、インデックス容量を増やすことなしに、ゾーン検索が可能になっている。また、(C)のタイプと比較した場合、全てのテキストでゾーンの開始位置と終了位置が等しいため、ゾーンの開始位置と終了位置の検索のためのオーバーヘッドを回避して、ゾーン検索が可能になっている。

図1にインデックス作成の手順を示す。図1のゾーンインデックスに示すように、オリジナルのテキスト上での位置に関わらず、全てのテキストに対してゾーンの「題」

*A large-scale text parallel search engine RetrievalExpress (2)structured text search methods

†Susumu Akamine, Toshikazu Fukushima, Yukiharu Yoneyama, Kiyosawa Haruhiko and Toshiyuki Tanaka
‡NEC Corp., NEC Informatic Systems Ltd.

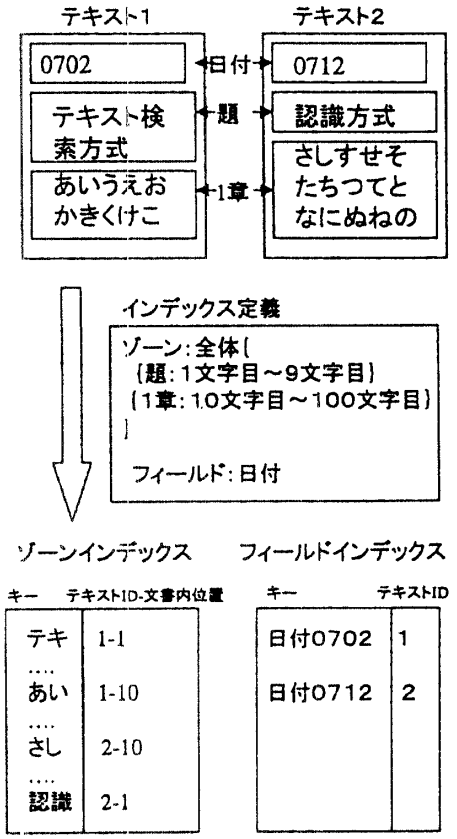

図 1: インデックスの作成手順

は1文字目から9文字目に、ゾーンの「1章」は10文字目から100文字目の位置情報をもつインデックスを作成する。これにより、検索時に検索結果のテキスト内位置より、該当のゾーンに対する検索結果かどうかを判定することで、特定のゾーンに対する検索を可能としている。

2.2 フィールド検索

フィールド検索は、フィールドにを示す情報をキー情報部にもたせることで実現した。例えば、図1のフィールドインデックスに示すように「日付」のキー「0724」に対しては、「日付0724」をキーとし、テキストIDを位置情報とするインデックスを作成することで、日付に対するインデックスを作成する。なお、RetrievalExpressのフィールド検索では、フィールドの完全一致／前方一致／範囲指定検索が可能である。

3 検索性能評価

特許公開公報の抄録テキスト(約80万件,570MB)を対象として、構造化テキスト検索方式の評価を行った。インデックスは、以下のように、2つに水平分割[3]を行い、各インデックスは複数のゾーン／フィールドをもつ形で構成した。

(1) 全文用インデックス
ゾーン:[発明の名称と要約、[発明の名称]、[要約]]

表 1: 検索対象と検索時間
評価マシン:NEC EWS4800/460,CPU:R10000

検索対象(ゾーン／フィールド)	検索時間(平均)
発明の名称	0.28秒
要約	0.38秒
発明の名称と要約	0.40秒
公開日(範囲検索)	0.10秒
発明の名称と要約 AND 公開日	0.45秒

(2) 日付用インデックス
フィールド:[出願日]、[公開日]

全文用インデックスのインデックス容量は1310MBであり、オリジナルテキストの約2.3倍で、ゾーンを使用しない場合のインデックス容量と同等であった。

検索対象ゾーン／フィールドと検索時間の関係を表1に示す。検索時間は、「発明の名称」・「要約」・「発明の名称と要約」の各ゾーンに対しては、「音声認識」・「機械翻訳」等の1単語からなる検索条件式の平均値であり、「公開日」のフィールド検索については、1993年1年間の範囲検索の値であり、「発明の名称と要約AND公開日」は、上の2つの検索条件の論理積の値である。

表1から分るように、ゾーンを用いた検索でも平均0.5秒以下の高速な検索レスポンスが得られており、この値はゾーンを用いない場合の検索時間とほとんど変わっていない。また、ゾーンとフィールドの複合検索では、RDMBSによるフィールド検索と全文検索エンジンを組み合わせたときに問題となるような極端なレスポンス低下はおきていない。

4 おわりに

大規模テキスト並列検索エンジンRetrievalExpressに実装した構造化テキスト検索方式を述べた。構造化テキストに対して、ゾーンとフィールドを用いた検索が高速に実現できることを確認した。今後、インデックスの水平分割とゾーン検索との組合せでの評価を行う予定である。

参考文献

[1] Salminen, A. and Tompa, F.W., "PAT Expressions: an algebra for text search," proc. of 2nd International Confernce of Computational Lexicography: COMPLEX'92, pp.309-332, 1992.
[2] 赤峯ほか、高速全文検索のためのフレキシブル文字列インバージョン法、情処ADBS、1996年。
[3] 福島ほか、大規模テキスト並列検索エンジンRetrievalExpress(1)並列検索方式、情処55全大、4N-07、1997年。