特徴ベクトルを用いた全文検索における高速化手法

3N－2

津田和彦†　青江順一††

†住友金属工業(株)　††徳島大学工学部

1．はじめに

文書作成のワープロ化やCD-ROMによる電子カタログ，電子辞書の出版など，文書の電子化が進んでいる．文書を電子化することにより生ずる利点は，印刷物と比較して意とするデータが高速に検出できるてんにある．しかし，更に大規模な文書データを電子化しようとした場合，高速な検索アルゴリズムが必要となる．このような状況の下，文書中よりあらゆる単語や文字列を高速に検索できる全文検索技術の必要性が重要視されてきた．

本論文では，全文検索技法の中で最も一般的である特徴ベクトルを用いた全文検索の高速化の一手法を提案する．

2．特徴ベクトル法

本章では，全文検索技法の中で最も一般的である特徴ベクトル法について，最も単純な方法の概要を説明する．

まず，図１に示すように検索対象となる文書を特定サイズのブロックに分割し，各ブロック毎に対応する特徴ベクトルを作成する．特徴ベクトルの作成方法は，対象となる文書ブロックに含まれる文字にある特定の算式を介し，特徴ベクトル中にその文字が含まれるか否かという情報を反映させる．例えば，下記特徴ベクトル作成法に示す方法などがある．

【特徴ベクトル作成法】
while（ブロック内の全文字）
　begin
　　Step1：　特徴ベクトルの全ビットを‘0’で初期化
　　Step2：　文字コード／特徴ベクトルサイズの剰余を求め，特徴ベクトルの剰余番目のビットに‘1’を立てる．
　end

実際にキーワードの検索を実行する場合は，検索キーワードに対しても同一方法で，同一サイズの特徴ベクトルを作成する．文書ブロックから作成した特徴ベクトルで，キーワードから作成した特徴ベクトルで‘1’の立つビットが‘0’となっている場合，その文書ブロックにはキーワードを構成する文字が含まれていないので，この文書ブロックにキーワードが含まれていないことがわかる．即ち，検索キーワードから生成した特徴ベクトルと，文書ブロックから生成した特徴ベクトルとでAND演算を行いその結果が，キーワードから生成した特徴ベクトルと同一ベクトルとなれば，その文書ブロックに検索キーワードが含まれる可能性があると判定できる．

このように，特徴ベクトルに対して検索を行うことでキーワードを含む可能性のある文書ブロックの絞り込みを行い，検索キーワードを含む可能性のある文書ブロックに対してのみ検索を実行する．即ち，特徴ベクトル法とは，特徴ベクトルを用いることで検索対象となる文書を絞り込み高速な検索を実現するアルゴリズムである．

3．高速特徴ベクトル法

前章で示したように，特徴ベクトル法では検索速度が向上するか否かは，特徴ベクトルの検索で検索対象となる文書ブロックをどの程度まで絞り込めるかにかかっている．

前章で示した【特徴ベクトルの作成方法】の場合，絞り込んだ文書ブロック中に検索キーワードが含まれていない原因は以下の２つが考えられる．

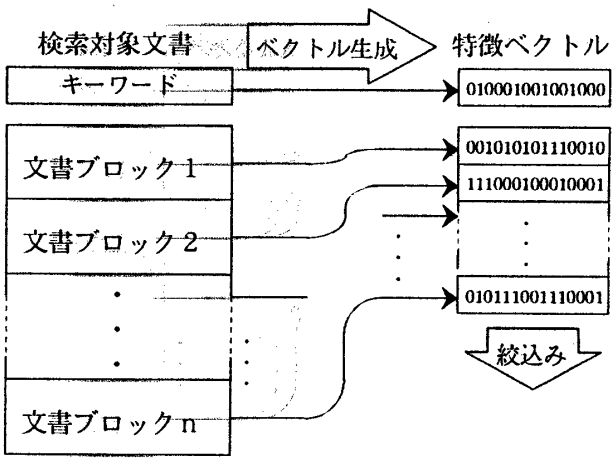

図１　特徴ベクトル法

The Fast Algorithm of Full Text Retrieval
Kazuhiko TSUDA † and Jun-ichi AOE ††
† Sumitomo Metal Industries, LTD.
†† The University of Tokushima

原因１：文字コード／特徴ベクトルサイズの剰余が同一となる文字が複数存在するため，文書ブロック中にキーワードを構成する文字が含まれていない．

原因２：文書ブロック中にキーワードを構成する文字が含まれているが，キーワード構成文字の位置，順序がキーワードと同一の列びになっていない．

そこで，本論文で提案する手法では，この原因のうち，原因１に示すものを取り除くことで，絞り込み率を向上させ高速化する方法を提案する．

原因１が起こる要因は，文字種類数に対してベクトルサイズが非常に小さいからである．よって，本手法では特徴ベクトルのサイズをコンピュータ上で扱うことのできる文字種類と同一サイズにする．

コンピュータ上で扱うことのできる文字種類は非常に多いと思われるが，JIS表記載の文字種類数は6,879であり，これにユーザ登録特殊記号などを追加してもせいぜい8,000種類程度を考慮すれば十分なサイズである．即ち，１文書ブロックに対応する特徴ベクトルサイズは，１KByte程度となる．

ここで，１つの文書ブロックサイズを一般的な特徴ベクトル法で用いられる文字数の範囲内である全角文字512文字とすると，その容量は１KByteとなる．即ち，特徴ベクトルの容量が文書全体の容量と同一となる程度である．

ここで問題となるのは，特徴ベクトルサイズが文書の容量と同一になるので，特徴ベクトル内の全ビットに対して絞り込みのAND演算を行えば，この演算に時間を要し，絞り込みを実行する意味がないことである．そこで，本手法では特徴ベクトルテーブルの演算を図２に示すようにビットスライス法を用いて実行する．ビットスライス法では，検索キーワードから生成した特徴ベクトルの‘1’の立つ列だけのAND演算を実行するだけで絞り込みが行える．このため，特徴ベクトルテーブルサイズが大きくなっても，演算対象となるデータ量を減少させることができる．

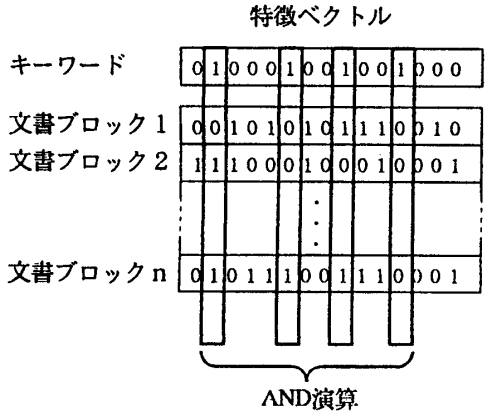

図２　ビットスライス法

表１　提案手法と一般的特徴ベクトルの比較

	提案手法	512	1024
ベクトル検索量	63K	6.3M	12.5M
文書検索量	0M	10M	1M
ベクトルサイズ	100%	6%	13%
正解率	100%	$(1/16)^5$	$(1/8)^5$

4．評価

本章では，本論文で提案した手法と２章で示した一般的な特徴ベクトル法とでの，特徴ベクトルの検索による絞り込み率を比較することによって評価する．

表１に評価結果を示す．文書サイズが100MByte，文書ブロックサイズは1KByte，検索キーワード長は５文字とした．また，一般的な特徴ベクトル法は特徴ベクトルサイズをを5124Bitと1024Bitの場合を用いた．

本手法のベクトル検索量が一般的な特徴ベクトル法と比較してよい結果がでているのはビットスライス法を用いたためである．このようにベクトルサイズが以下に大きくとも，ビットスライス法を用いることで検索すべき特徴ベクトルの量を少なくすることが可能である．文書検索量は特徴ベクトルの検索による絞り込み率と比例する．提案手法は検索キーワードの文字種による絞り込みが完全となるので，検索すべき文書ブロック数は非常に少なくなる．正解率とは，絞り込みされた文書ブロックに検索キーワードを構成する全ての文字が含まれる確率である．本手法は全文字種に対する特徴ベクトルを用いるので100%となるが，ベクトルサイズ512Bitの場合，文字種に対するビット幅が1/16しかないので，１つのビットで平均16種類の文字が含まれる割合となる．

5．おわりに

本手法の欠点としては特徴ベクトルサイズが一般的な特徴ベクトル法と比較して非常に大きくなる点が上げられる．しかし，評価に用いたデータの場合文書ブロックと同一のサイズであり，近年のＨＤの大容量化などを考えると致命的なものではない．

今後は，本手法の実験を進め，実際にどの程度高速な検索が可能であるかを検証する．

参考文献

(1)William B. Frakes, Ricardo Baeza-Yates : "Infomation Retrieval, Data Structures & Algorithms", Prentice Hall 1992.

(2)菊池忠一: "日本語文書用高速全文検索の一手法", 信学論D-I ,Vol. J75, 9, PP.836-846, 1992.