WWW 全文検索システム Verno のデータベース

4T－6

沼尻務　竹岡厚　渡辺高志　田川信一　上田和紀

早稲田大学大学院理工学研究科情報科学専攻
早稲田大学理工学部情報学科

1　はじめに

今日、WWW 上ではさまざまなテキストデータが膨大に存在し、それにともない検索エンジンの役割も重要さを増している。の本研究では、ユーザの複雑かつ多岐に渡る要求に対処しうる WWW 全文検索システム Verno[1] の設計と実装を行なっている。

2　Verno のデータベースの特徴

Verno のデータベースでは全角文字を N-gram[2] 方式により処理している。これは、文の一部など任意の文字列を検索できる、完全照合検索だけでなく部分一致検索や最適照合検索にも適しているなど、より柔軟な検索が行なえるからである。Verno では、出現頻度の高い平仮名や片仮名は 3-gram、それ以外の字種は 2-gram で処理を行なっている。また、半角文字は'.' や',' など、区切り文字で区切られた連続する文字列のうち、先頭と末尾が英数字である最長のものを検索単位 (以後単語と呼ぶ) としている。これは、'TCP/IP' のような記号を含む文字列も検索できるようにするためである。

ファイルの先頭からのバイト数 (位置情報と呼ぶ) も共にデータベース化することにより、N-gram 方式の場合におこりうる誤検索を防ぎ、検索したキーワードが出力結果のファイルに必ず存在することを保証する。

3　データベースの構成と検索方法

3.1　単語と URL アドレスの扱い

EUC コードの全角文字の第 1 バイト、第 2 バイトをそれぞれ a_1, a_2 としたときに $A = (a_1 - 163) \times 94 + (a_2 - 161)$ とする。連続する 3 文字の値をそれぞれ A_1, A_2, A_3 としたとき、3-gram の単語において $T = -(A_1 \times 1,000,000 + A_2 \times 1,000 + A_3)$ で計算される値、また 2-gram の単語において $B = A_1 \times 10,000 + A_2$ で計算される値は全て一意に定まり、かつ 32bits で表せる。そこで、この値を単語に代わって扱うことにより処理をしやすくしている。また、半角文字の単語では MD5[3] の値を利用している。(以後、単語に付けられた値を wordID と呼ぶ。)

また、検索結果として返す URL アドレスはそのままでは扱いにくいので、各 URL アドレスに一意の番号である URLNo を割り振っている。この番号の割り振り方は、同一サーバに存在するファイルには近接した番号を与えるようにしている。これにより検索結果をサーバごとに分類するなどといった検索方法を容易に行なえるようにしている。

3.2　検索の流れ

データベースは主に以下の 4 つの部分から構成されている。

1. ハッシュ表: wordID と該当するデータ参照ファイルのアドレスがバケットに入っている。ハッシュを使うのは、利用されていない wordID が数多くあること、高速に検索処理を行なうため主記憶装置上に表を格納できるようにするためである。

2. データ参照ファイル: 2-gram,3-gram, 半角文字用に各 1 つずつ存在する。各単語の出現したファイル数、頻度データファイル内のデータ該当位置、位置情報ファイルのデータ該当位置をデータとして保有している。

3. 頻度データファイル: URLNo とファイル内の出現頻度が 1 つのデータとして、単語毎に URLNo の順に二次記憶装置内に置かれている。2-gram では wordID の上位 12bit の値により、また 3-gram は 40,000,000 ごとに分割されている。一方、半角文字の単語については、半角文字で 1 つのファイルに置かれている。

4. 位置情報ファイル: 単語の位置情報が単語毎に、URLNo 順に二次記憶装置内に頻度データファイルと同じ方法で分割して置かれている。頻度データファイルと位置情報ファイルが分かれているのは歴史的な事情による。

Database of Verno – A WWW Full-text search system.

Tsutomu Numajiri, Atsushi Takeoka, Takashi Watanabe, Shin'ichi Tagawa, Kazunori Ueda.

Dept. of Infomation & Computer Science Course, Graduate School of Science & Engineering, Waseda University.

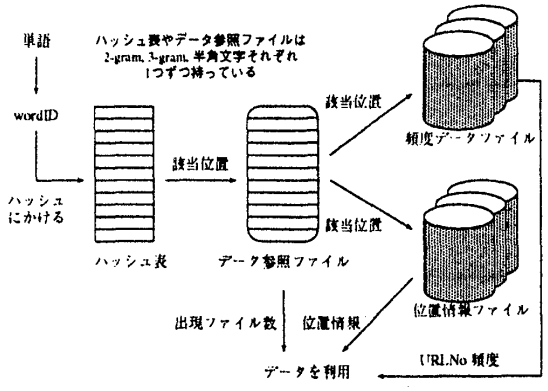

図 1: 検索の流れ

データベース検索

キーワードを入力した際のデータベース検索の流れは図1のようになる。

1. 単語の wordID を 3.1節の方法により求め、さらにその wordID をハッシュにかける。ハッシュ値は wordID のハッシュ表のバケット数による剰余を用いる。
2. (1) より出現ファイル数、頻度データファイル内のアドレス、位置情報ファイル内のアドレスを取得する。
3. URLNo とファイル内の出現頻度を出現ファイル数分取得する。取得したデータは URLNo 順にソートされている。
4. 取得したファイル内の出現頻度の分だけ位置情報を取得する。取得したデータは URLNo 順かつファイルにおける出現順にソートされている。

また検索するキーワードが複数の単語にわかれる時には、各単語によって検索を行なう。次に返された集合から URLNo が一致するものを取り出す。さらにキーワードを構成する単語の位置の差分と、各検索候補の位置情報の差分が一致するものを検索結果とする。(これを全単語走査と呼ぶ。)

これに対し、ある単語の出現ファイル数がその両隣の単語の出現ファイル数よりも小さいもの及び末尾の単語のみを使って絞り込む方法も実装している。(これを部分走査と呼ぶ。)

4 性能評価

PentiumII 300MHz PC において、ロボットが収集し、全て日本語コードを EUC に変換したテキストファイル 910,857 ファイルに対してデータベースの構築を行ない、その構築にかかる時間を計測した。出現した単語の数は表1のようになり、構築には 70,411[s] 要した。

表 1: 単語の数

種類	2-gram	3-gram	半角文字
のべ数	185,095,793	96,091,182	27,437,744
種類数	1,958,192	707,812	2,499,762

また、作成したデータベースに対して検索にかかる時間を計測した。キーワードを単語に分割し、検索工程 (2) が終わるまでの時間 (a) と、検索工程 (3) 以降、絞り込みにより検索結果が出るまでの時間 (b) を全単語走査及び分走査の 2 方法で計測した結果が表 2である。

表 2: 検索時間の比較

全単語走査			
入力キーワード	(a)[μs]	(b)[μs]	検索数
早稲田	542	12,021	3,991
早稲田大学	963	47,100	2,713
早稲田大学理工学部	1,743	85,021	414
早稲田大学理工学部情報学科	1,676	81,410	36
アプリケーションプログラム	2021	117,824	265
部分走査			
入力キーワード	(a)[μs]	(b)[μs]	検索数
早稲田	529	12,726	3,991
早稲田大学	884	48,564	2,713
早稲田大学理工学部	1,151	24,262	414
早稲田大学理工学部情報学科	1,687	17,114	36
アプリケーションプログラム	1,684	18,013	267

全単語走査に対して部分走査は最大で約 1/6 に所要時間を短縮することができた。また全ての単語を絞り込みに用いていないことから起こる誤検索も極めて少ないことがわかった。

5 まとめ

現在までに、実用に耐え得るデータベースの構築及び検索が可能になった。しかし、ユーザに質の高い検索結果を提供するための処理を考えると、データベースもさらなる高速化を模索する必要がある。また、キーワードに対して完全一致検索だけでなく、部分一致検索や最適照合検索もできるように実装を行なっていく。

参考文献

[1] http://verno.ueda.info.waseda.ac.jp/

[2] 長尾真編: 自然言語処理, 岩波講座ソフトウェア科学 15, 岩波書店, 1996.

[3] R.Rivest, RFC1321: The MD5 Message-Digest Algorithm, 1992.

[4] 田川信一他: 学習型 WWW 検索エンジン Verno, 情処学 56 全大, 6Z-07, 1998.