圧縮ファイルに直接検索を行なう一手法*

5 L − 5

多々納 勉　　大塚 真吾　　宮崎 収兄

千葉工業大学工学部情報工学科

1 はじめに

現在オフィスや家庭にコンピュータが普及し, インターネット利用者の増加により電子メールやホームページなどの電子文書の持つ情報の重要性が高くなってきた. しかし電子文書を管理保存するためにはコンピュータが持つ記憶媒体の圧迫が問題となる. そこで文書サイズの縮小のために文書ファイルに対して可逆変換圧縮をおこない管理をする. しかし圧縮された文書は符号化されているため, その内容を直接閲覧, 文字列検索することは困難である. 検索するためには復号を行ない文書全体を元に戻し, その上でさらに検索をしなければならない. このようなオーバーヘッドを削減しようとする研究は写像を利用した圧縮法を用いた $2^n(k)$ 符号語 [1] などがある.

我々は完全な復号を行なわずに, 任意の文字列の検索を行なう新たな検索法を提案する. 動的局所辞書を用いたLZFG符号をベースとして圧縮したファイルに対する文字列検索を行なう手法を示す.

2 LZFG符号化と直接検索

LZFG符号化法 [2] はZivとLempelによって提案されたユニバーサル符号化法 [3] の数多くの変形の一つで高い圧縮効果と1パスの圧縮法であるため高速性を合わせ持った符号化法である. この手法はスライド窓を利用し作成される辞書を用いた符号化法で, トライを一般化したパトリシアで辞書表現し早く最長一致文字列を探索でき, またパトリシアの木構造を利用して高い圧縮効果が期待される. LZFG符号は有限状態情報源については, 記号列長が長くなるにつれて平均符号長がエントロピーに収束するような理想的な符号化が行なえることを明らかにしている [4].

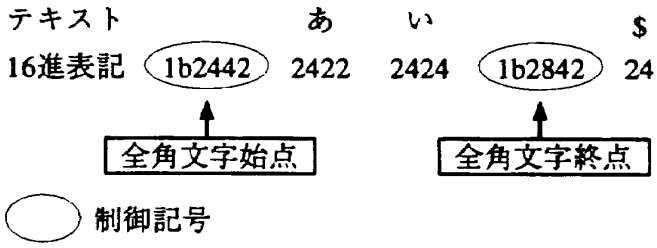

図 1: EU コード

LZFG符号では出力ファイルに書き込まれるのは参照文字列の一致文字列長と位置や未登録文字だけである. そのため目的の文字列を検索しようとするとファイルの最初から復号を行ない, 検索文字列を探索しなければならない. このため直接検索をする場合には圧縮ファイルに復号しなくてもわかるキーワードを埋め込む必要がある. キーワードとしてもっとも適切なものはやはり元の文書の一部である.

そこで英文テキストを対象として,LZFG符号化の符号化最小単位を8ビットにとり, 圧縮したファイルから元の文書の検索文字列の1語を探索するものを実装したところ, 圧縮効果は良好であった. しかし日本語文書を対象とするためには, 半角文字の探索は行なえるが全角文字の探索には問題がある. 例えば図1での"$"と全角文字の1バイトが同一のものとして扱われてしまう.

3 日本語検索を可能にするLZFG符号化

扱うファイルがEUコード(EUC)の日本語テキスト文書である場合, 図1のように全角文字は1語16ビットで半角英数字は1語8ビットで表される. またEUCは全角文字を表すのに始点と終点にそれぞれ3バイトの制御記号を置くようになっているため全角文字と半角文字の区別がつけられる.

2節の方法では日本語を扱う際に以下の問題点が挙げられる.

*A Method for Direct Search of Compressed Files by Tsutomu Tadanoh and Shingo Ohtsuka and Nobuyoshi Miyazaki (Chiba Institute of Technology)

1. 1語8ビットで圧縮を行なうと全角文字の上位8ビットと下位8ビットが別れて辞書に登録されてしまう.

2. 全角文字を表している上位,下位ビットはそれ単体で半角文字を表していることがあるため辞書に登録された文字列は本来存在しない文字が登録される可能性がある.

3. 圧縮ファイルに対する検索を行なう際にパトリシアを再構成しているが,パトリシアを完全に再構成するために処理時間が期待するよりもかなり長くなってしまった.

これらの問題を解決するため以下のような改良手法を提案する. 1番目と2番目については辞書登録とファイルへの書き出しの際に全角文字の上位8ビットと下位8ビットが別れて登録されないように,登録される文字が全角文字の上位8ビットなら続く下位8ビットも辞書に登録しファイルに出力するように条件を加える. 3番目の問題は検索文字列の各語をそれぞれファイルの始めから探索し, 見つかったところからのスライド窓中の位置関係だけを利用すればより早く検索が可能になる. しかしこの場合,探索の時間は検索文字列の長さに比例して大きくなるため, 検索文字列は例えばある単語のような短い文字列である必要がある.

4 圧縮ファイルに対する検索法

LZFG符号のように動的に変化する局所辞書を利用して符号化を行なう手法では, 入力に対して出力が一意に決定する写像を用いた符号化法と比べて圧縮ファイルに対する文字列検索は困難となる.

図2のように写像を利用すると検索文字列を写像式によって変換し,変換された出力文字列を探索すれば良いが,動的局所辞書を用いる方法では辞書に含まれる参照文字列の位置関係で出力が変化するため,前者のような方法はとれない. そこで動的な局所辞書の性質を利用した検索法を述べる.

LZFG符号化を行なうとファイルには符号語が大別して以下の2つのモードで出力される.

- 直接モード スライド窓中に存在しない文字そのものの出力
- 間接モード スライド窓中に存在する一致文字列の位置と一致長の出力

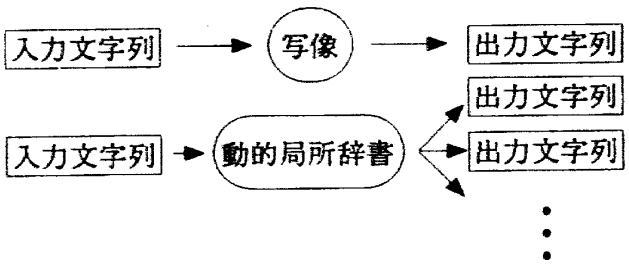

図 2: 圧縮アルゴリズム

そのためにパトリシアに登録されるには先に直接モードによって新たな語が出力される必要がある. したがって,圧縮ファイルから検索文字列の一部かどうかを検証するためにパトリシアを再構成しつつ,検索文字列の文字が最初に現れる場所を探す. その際にEUCの制御文字を発見したら全角半角の判別を行なうフラグを立てる. そして検索文字列の一部を発見した場合でフラグの状態が検索文字列の形式と一致したら,再構成されたパトリシアの木構造からその探索された文字列を復号し,検索文字列の一部なのかを判別する.

5 おわりに

動的局所辞書を用いたLZFG符号ベースとして圧縮されたファイルに対する単純な文字検索を実装した. そして日本語テキストに適応する際の問題点を挙げ, それを解決する新たな手法を提案した.

参考文献

[1] 松本 光崇, 角田 達彦, 松本 裕治: 圧縮ファイルへの直接検索を可能にする符号化法の提案, データベースシステム,vol. 107,No. 6,pp. 41-48 (1996)

[2] Fiala,E. and Greene,D.: Data Compression with Finite Windows, *Comm. ACM*. Vol. 32, No. 4,pp. 490-505 (1989).

[3] Ziv,J. and Lempel,A.: A Universal Algorithm for Sequential Data Compression, *IEEE Trans. Inf. Theory*,Vol 23, No. 3,pp. 337-343 (1977).

[4] 森田 啓義, 小林 欣吾: 制約つき再生可能な文字列分解にもとずく計算機ファイルのデータ圧縮, 情報処理学会論文誌,Vol. 33,No. 2,pp. 110-121 (1992)