Mem. Fac. Educ. Ehime Univ.,Nat. Sci., Vol. 14, No. 1 63～69 (1993)

Ziv-Lempel符号による日本語文書データ圧縮法の性能向上

稲 井 義 正
（愛媛大学教育学部技術科教室）

（平成5年4月26日受理）

Improvement of Zip-Lempel Family Encoders for Japanese Text Compression

Yoshimasa INAI
Department of Technology, Faculty of Education,
Ehime University, Bunkyo-cho, Matsuyama 790, JAPAN

(Received April 26, 1993)

The performance of any data compression method is highly dependent on the data being compressed. This paper descrides a simple and useful code of Kanji characters, which is suitable for doing data compression of Japanese text.

The experimental results shows that the performance of Ziv-Lempel family encoders are improved using this Kanji code.

1. まえがき

ユニバーサル符号化法を適用すれば，対象データ（情報源）の統計的な性質を前提とはしないで，適応的にデータ圧縮（情報源符号化）が行える．ユニバーサル符号のひとつであるZiv-Lempel符号は，理論には漸近的最良性を満たす強力な符号であるが[2]，実際のデータを圧縮する場合には，圧縮率，集束性，計算量の面ではそれほど実用的なものとはいえない．漸近的最良性は示されていないが，より実用的な符号化法に，Lempel-Ziv-Welchによる方法（LZW法）がある．

筆者は文献(1)において，LZW法を適用する対象を日本語で書かれた文書に限定することで，圧縮率や処理時間の面で性能が改良できることを示した．その改良法として，圧縮する単位と

して平仮名や漢字を1文字として扱い，JIS第一水準の文字を16ビット符号ではなく，より短い符号語長の12ビットで表現する日本語文字の符号化法を提案した．しかし，その符号には，

(1) 符号語長が12 bitであるから4096字を表現できるが，そのうち約3500の符号しか使用していない．

(2) 符号化する第一水準の漢字文字にはほとんどデータの中に出現しないと思われる文字がある（例えば「壓」や「傴」）．

等の点でまだ冗長性が残っている．この冗長性を減少させれば，更に圧縮率が改善されるはずである．

本論文で提案するのは，このふたつの冗長性を共にある意味で減少させた日本語文字の符号化法である．また，その日本語符号を使用することで，圧縮率および実行時間が改善されることを実験結果により示す．

2．日本語文字の符号化

以下，簡単のため，誤まる恐れのない場合には，本論文で提案する文字の符号に含まれる文字を（平仮名なども含むが）漢字，その符号を漢字符号，またそれ以外の文字を非漢字（8ビットで表す），その符号を非漢字符号と書く．

さて文献(1)の漢字符号に含まれる冗長性を減少させるには，例えば，符号を12ビットの固定語長ではなく，いわゆる可変語長の符号にすることが考えられる．そのように符号を可変語長にすることは，圧縮処理出力の符号に対して行えばよいので（圧縮率を改善する方法であるが，実行時間は増加する），本論文では，日本語文字を固定語長で表すことにする．

また，冗長性の(1)だけを減少させたいならば，例えば，使用していない符号に新しく文字を割り当てることが考えられるが，これについては，別の機会に報告する．

文献(1)では，JIS第一水準の文字の中から約3500文字を漢字として取り扱ったが，本論文では更に漢字として取り扱う文字を少くすることを考える．

表1に日本語文書の中に現れる各文字の割合を求めた結果を示す．

いま，文字の出現頻度などの統計量を利用せずに，第一水準の文字数より少ない漢字を選ぶとすれば，常用漢字（1945字）あるいは教育漢字（1006字）がその候補としてでてくる．しかし，表1からも分かるように，日本語文書には，間隔，平仮名（83字），カタカナ（86字），句点や読点あるいは記号などの文字が多く使用されている．つまり，これらの文字のうち最低でも約170字を符号化する文字の中に含める必要がある．そうした場合における文字の総数を計算すれば，常用漢字を選べば2048字（11ビット）を，また教育漢字を選べば1024字（10ビット）を少しだけ超過する．符号化する字数は2のべき乗とするのが都合よく，これらの文字数は冗長性を減少するには不適当な値である．従って，両方とも漢字符号の候補

表1 文字の出現回数

内訳	出現回数	比率
平仮名	605674	31.7［%］
カタカナ	331583	16.8［%］
間隔（空白）	85518	4.5［%］
英字	63882	3.3［%］
数字	28117	1.5［%］
句点（。）	26545	1.4［%］
読点（、）	25910	1.4［%］
漢字	633242	33.1［%］
記号	122030	6.4［%］

＊2バイト文字の全文字数 1912501
＊比率は2バイト文字に対する値
＊全データの大きさ 6864051[byte]
＊全文字数 4951550

とすることはできないので，何らかの統計処理を必要とする．

さて，文献(1)のプログラムを利用した予備実験の結果から次のふたつの事が分かっている．文書に使用されている異なった漢字の字数はほとんどが1000以下であった．また，出力される漢字符号の総数から推定すると，漢字符号の長さ1ビット当りおよそ1パーセントの割合で圧縮率を改善できることがわかった．従って，圧縮率の改善効果を大きくするために，漢字符号のビット数は10ビット以下としたい．これは，漢字符号化する文字の数を1024字以下とすることを意味する．

そのため，データ全体で約6.5 Mbyte となる日本語文書を解析し，文字の出現頻度を求めた．その結果を利用して，文字の出現頻度の多い順に1024字，512字及び256字を選び，漢字として符号化するプログラムを作成した．プログラム内部の符号として，非漢字文字を0から255までの値で，漢字文字を256から最大値で1279（1023＋256）までの値として表している．

出力では漢字と非漢字を区別している．

3．圧　縮　実　験

この1024，512あるいは256字に漢字を制限することが，データ圧縮にとって有効な符号化法であることを実証するために計算機による実験を行った．

本論文で日本語文書の圧縮実験に使用した方法は，LZW[3]（Lempel-Ziv-Welch）法，LZSS[4]（Lempel-Ziv-Storer-Szymanski）法，およびLZW 法の改良版に相当する Storer による方法[5]（そのプログラムの名前から SQUEEZE 法と呼ぶことにする）である．

ここで圧縮率は次式で定義する．

$$\text{圧縮率}=\frac{\text{圧縮後のデータのバイト数}}{\text{圧縮前のデータのバイト数}}\cdot 100\ [\%]$$

表2に文書の大きさなどの値を例として示す．圧縮実験は約100個の文書を対象として行った．

各方法の名前の最初に 'K' をつけて漢字符号を使用した方法であることを表し，末尾の数字が漢字符号の符号語長を表している．符号語長12である方法は，JIS 第一水準の文字を漢字と見なした場合を表す（KLZW-12 は文献(1)では K-LZW1 法と呼んだ）．

表2　実験に使用した文書データの例

文書	大きさ [byte]	漢字の出現回数（注1）	全文字数（注2）
A	4337	1884(86.9[%])	2453(56.6[%])
B	5937	2587(87.1[%])	3350(56.4[%])
C	7642	2034(53.2[%])	5608(73.4[%])
D	10129	3533(69.8[%])	6596(65.1[%])
E	12119	3898(64.3[%])	8221(67.8[%])
F	14303	3584(50.1[%])	10719(74.9[%])
G	16395	5356(65.3[%])	11039(67.3[%])
H	20430	1127(11.0[%])	19303(94.5[%])
I	22528	4814(42.7[%])	17714(78.6[%])
J	24816	10787(86.9[%])	14029(56.5[%])
K	31041	9730(62.7[%])	21311(68.7[%])
L	35925	7484(41.7[%])	28441(79.2[%])
M	40069	17646(88.1[%])	22423(56.0[%])
N	46471	14928(64.2[%])	31543(67.9[%])
O	55235	24142(87.4[%])	31093(56.3[%])
P	63124	17630(55.9[%])	45494(72.1[%])
Q	81258	31110(76.6[%])	50148(61.7[%])
R	117480	37783(64.3[%])	79697(67.8[%])
S	150295	14213(18.9[%])	136082(90.5[%])
T	251904	21660(17.2[%])	230244(91.4[%])

注1．（）内は漢字のバイト数での割合
注2．（）内は文字数／バイト数

3．1　圧縮率の実験結果

3．1．1　LZW 法

インプリメントの詳細は文献(1)と同様である．新しく登録する文字列に使用する辞書を同じ大きさとして比較実験を行った．表3が LZW 法の実験結果である．表3(a)は辞書の大きさを4096としたときの圧縮率である．表3(b)は全データに対する圧縮率の平均値，最大値および最小値を示して

いる．KLZW-12 と KLZW-10 の圧縮率の差約2パーセントは，前章で述べた改善率の推定値とほぼ等しい．すなわち，漢字符号の語長1ビット当り約1パーセント圧縮率は改善されている．

3．1．2 LZSS 法

LZSS 法には，圧縮済みの入力文字列をそのまま記憶するためのバッファ（辞書）が存在する．圧縮はそのバッファ内で文字列の最長一致系列を検索することで行う．もし適当な長さ以上の系列が見つかればその位置と一致長を符号化して出力し，見つからなければ文字符号そのものを出力する．位置を12ビット，一致長を4ビットで表すとした場合，実験の結果，漢字符号を使用する場合は一致長は2から17まで，漢字符号をしない場合は3から18までの値とするのが適当であった．表4に実験結果を示す．表4(a)は圧縮率の例，表4(b)は実験した全ての文書に対する平均値，最大・最小値である．

3．1．3 SQUEEZE 法

この方法は種々ある LZW 法の改良版のひとつと見なせる．この方法では辞書に登録された文字列の利用効率を改良している．すなわち，LZW 法ならば辞書が飽和したならそれ以上新

表3 LZW法による圧縮実験の結果
（辞書の大きさは4096）

(a) 各方法による圧縮率の例

文書	LZW [%]	KLZW-12 [%]	KLZW-10 [%]	KLZW-9 [%]	KLZW-8 [%]
A	52.0	43.2	39.4	38.3	38.7
B	58.2	50.2	46.8	46.8	48.0
C	55.8	50.7	48.2	48.1	49.0
D	57.6	50.5	47.9	47.4	48.0
E	52.6	45.3	43.7	43.4	43.8
F	46.6	39.8	37.8	37.6	38.2
G	39.6	33.2	32.1	33.0	33.6
H	55.8	55.3	54.9	56.8	57.1
I	21.5	15.7	15.3	16.2	17.2
J	50.3	41.4	39.3	38.9	39.8
K	58.8	52.2	49.9	50.2	50.8
L	36.6	31.4	30.4	30.3	30.8
M	60.4	49.7	47.6	48.7	50.5
N	54.9	46.5	44.6	44.7	46.1
O	54.4	42.5	40.6	40.9	42.2
P	50.0	43.3	41.4	41.4	42.2
Q	51.6	43.2	41.2	41.2	42.2
R	62.8	54.9	53.5	54.5	55.7
S	65.9	65.7	64.6	64.7	65.6
T	15.1	13.9	13.6	13.7	13.9

(b) 全文書に対する圧縮率

	LZW [%]	KLZW-12 [%]	KLZW-10 [%]	KLZW-9 [%]	KLZW-8 [%]
平均値	51.0	44.2	42.3	42.3	43.3
最大値	70.1	68.3	66.4	67.1	68.4
最小値	9.4	6.6	6.5	6.7	7.5

表4 LZSS法による圧縮実験の結果
（辞書の大きさは4096）

(a) 各方法による圧縮率の例

文書	LZSS [%]	KLZSS-12 [%]	KLZSS-10 [%]	KLZSS-9 [%]	KLZSS-8 [%]
A	43.0	39.9	36.6	35.3	35.3
B	52.5	48.8	45.7	45.5	46.2
C	49.3	47.3	44.9	44.5	44.9
D	49.9	46.9	44.5	43.9	44.4
E	41.2	37.8	36.3	35.9	36.0
F	38.3	36.8	34.9	34.4	34.7
G	34.0	31.3	30.2	30.6	31.0
H	58.9	56.4	55.9	57.8	59.1
I	20.8	19.1	18.3	18.5	19.0
J	42.7	38.0	36.0	35.7	36.2
K	50.6	47.5	45.4	45.4	45.9
L	30.7	29.0	27.7	27.5	27.9
M	52.4	46.8	45.0	45.5	46.8
N	44.7	41.5	39.6	39.3	40.0
O	39.5	35.3	33.5	33.2	33.8
P	39.7	37.5	35.8	35.4	35.7
Q	42.1	38.1	36.4	36.2	36.8
R	40.1	36.5	35.3	35.8	36.5
S	41.9	41.2	40.4	40.5	40.9
T	22.4	21.8	21.2	21.3	21.5

(b) 全文書に対する圧縮率

	LZSS [%]	KLZSS-12 [%]	KLZSS-10 [%]	KLZSS-9 [%]	KLZSS-8 [%]
平均値	42.3	39.2	37.5	37.3	37.9
最大値	62.9	62.1	60.7	60.8	61.7
最小値	16.3	15.6	15.2	15.2	15.5

しい文字列が登録できなくなるのに対して，SQUEEZE 法では，その後使用されないと思われる文字列を決められた方法で辞書から削除する．また一度に登録する文字列を LZW 法に比べて，多くあるいは長くしている．実験は辞書の登録削除法のうちから AP-LRU (all prefixes, least recently used) 法で行った．

この方法も LZW 法と同じく，新しく登録する文字列用の辞書の大きさを同じ値にして比較した．表5に結果を示す．表5(a)は圧縮率を示す．表では 'SQUEEZE' を短く 'SQZ' と表記した．全データに対する結果の平均，最大および最小値は表5(b)のとおりである．ここでも，漢字符号の語長1ビット当たり約1パーセントの圧縮率の改善が見られる．

3.2 圧縮に要する実行時間

全ての方法で圧縮に必要とされる時間を計測した．その結果，LZW 法及び SQUEEZE 法ではほぼ入力データの大きさに比例する時間が必要であった．また，LZSS 法では，同じ大きさであってもデータによって必要な時間は他の方法よりも大きく変化するが，平均値はほぼ入力データの大きさに比例した．そこで，表6では実行時間を入力データの1バイト当りに要

表5 SQUEEZE法による圧縮実験の結果
(辞書の大きさは4096)

(a) 各方法による圧縮率の例

文書	SQZ [%]	KSQZ−12 [%]	KSQZ−10 [%]	KSQZ−9 [%]	KSQZ−8 [%]
A	45.9	41.1	37.5	36.3	36.3
B	53.8	48.0	44.7	44.7	45.5
C	51.1	48.0	45.4	45.2	45.8
D	52.8	48.0	45.3	44.7	45.2
E	45.6	40.2	38.5	38.1	38.5
F	40.4	37.7	35.5	34.9	35.3
G	36.9	32.0	30.6	31.0	31.4
H	51.8	48.8	48.5	50.2	51.6
I	16.5	13.1	12.7	13.0	13.8
J	46.7	39.1	36.7	36.3	37.1
K	53.1	48.0	45.6	45.5	46.1
L	30.6	28.1	26.6	26.5	26.9
M	56.1	46.8	44.6	45.5	47.1
N	47.3	41.6	39.3	38.9	39.9
O	44.4	36.8	34.7	34.3	35.2
P	41.3	37.0	34.9	34.5	35.0
Q	45.6	39.2	37.1	36.7	37.6
R	42.7	37.9	36.7	37.5	38.2
S	42.1	40.5	39.7	39.8	40.4
T	14.6	13.5	13.0	13.2	13.5

(b) 全文書に対する圧縮率

	SQZ [%]	KSQZ−12 [%]	KSQZ−10 [%]	KSQZ−9 [%]	KSQZ−8 [%]
平均値	44.1	39.2	37.2	37.0	37.8
最大値	62.2	61.2	59.9	59.8	60.1
最小値	8.9	6.9	6.6	6.8	7.2

表6 圧縮に要する実行時間

方法	実行時間 [sec/byte]	比率1	比率2
LZW	0.780E−4	1.000	1.000
KLZW−12	0.702E−4	0.900	0.900
KLZW−10	0.827E−4	1.060	1.060
KLZW−9	0.833E−4	1.068	1.068
KLZW−8	0.854E−4	1.095	1.095
LZSS	3.965E−4	1.000	5.083
KLZSS−12	2.934E−4	0.740	3.762
KLZSS−10	3.051E−4	0.769	3.912
KLZSS−9	3.089E−4	0.779	3.960
KLZSS−8	3.163E−4	0.798	4.055
SQZ	1.963E−4	1.000	2.512
KSQZ−12	1.562E−4	0.796	2.003
KSQZ−10	1.693E−4	0.862	2.171
KSQZ−9	1.713E−4	0.873	2.196
KSQZ−8	1.770E−4	0.902	2.269

＊比率1：各方法で漢字符号を使用しない場合と比較
＊比較2：LZW法を基準にして比較

した時間で表した．比率1は各方法別に時間の比を計算した値，比率2はLZW法を1として計算した値である．

3．3 実験結果のまとめ

(1) 本論文で提案した日本語文字の符号化法は，実験した3種類のデータ圧縮法全てに有効である．

(2) 漢字として取り扱う文字を1024あるいは512に制限した場合が圧縮率が最もよい．しかし両者にはほとんど差はない．このときの，実行時間はLZSS法およびSQUEEZE法では少しだけ漢字符号を使用しない場合より短くなる．

(3) 圧縮率は，LZSS法とSQUEEZE法は平均値ではほとんど同じ値であり，LZW法はそれに比較して少し悪い．

(4) 実行時間は，LZW法以外は漢字符号の語長が大きいほど短くなる．

(5) LZW法に比べてLZSS法が4倍，SQUEEZE法が2倍の時間が必要である．

4．考　　察

4．1 最適な漢字符号語長

実験結果より分かるように1024（10ビット漢字符号）または512（9ビット漢字符号）字の日本語文字を漢字として符号化すれば最適である．

これは次の理由によると思われる．表7はデータとした文書に使用されている異なる日本語文字の数と，その内で制限した漢字符号に含まれる文字の数を示している．この表より，漢字符号語長が10ビットの場合には，含まれる割合は平均して約90パーセントであり，文書に現れるほとんどの日本語文字が漢字符号に含まれていることが分かる．これは，固定語長の漢字符号としてほぼ無駄がないことを意味する．また，漢字符号語長が9ビットの場合には，平均して約70パーセントしか含まれないので，固定語長の符号としては少し無駄がある．しかし，10ビットに比べて1ビット少ない分だけ漢字を短い符号で出力できることがこの無駄を補い，10ビットの場合とほとんど圧縮率が同じ値となったと思われる．

また，8ビット漢字符号の場合は，平均して約44パーセントしか漢字符号に含まれていないので，その符号の半分しか使用されず無駄が多い．

表7 符号語長を変化した場合に漢字符号に含まれる日本語文字の字数

(a) 結果の例に対する値

文書	全字数	10ビット	9ビット	8ビット
A	241	240(99.6)	222(92.1)	178(73.9)
B	366	328(89.6)	260(71.0)	171(46.7)
C	313	303(96.8)	258(82.4)	182(58.2)
D	347	336(96.8)	292(84.2)	194(55.9)
E	306	293(95.8)	252(82.4)	176(57.5)
F	357	344(96.7)	300(84.0)	206(57.7)
G	262	239(91.2)	189(72.1)	134(51.2)
H	1127	880(78.0)	507(45.0)	254(22.5)
I	226	219(96.9)	184(81.4)	129(57.1)
J	510	481(94.3)	378(74.1)	230(45.1)
K	619	556(89.8)	396(64.0)	228(36.8)
L	466	443(95.1)	357(76.6)	227(48.7)
M	676	588(87.0)	392(58.0)	223(33.0)
N	645	584(90.5)	419(65.0)	233(36.1)
O	587	553(94.2)	430(73.3)	246(41.9)
P	660	601(91.1)	441(66.8)	240(36.4)
Q	710	653(92.0)	455(64.1)	250(35.2)
R	1044	744(71.3)	427(40.9)	231(22.1)
S	968	738(76.2)	453(46.8)	243(25.1)
T	1381	802(58.1)	422(30.6)	213(15.4)

＊全字数：文書に現れる異なる2バイト文字の数

＊( )内は全字数に対する割合［%］

(b) 全文書に対する値

	全字数	10ビット	9ビット	8ビット
平均値	576	482(90.6)	350(69.6)	210(43.8)
最大値	2923	916(99.7)	507(96.6)	255(79.1)
最小値	148	143(31.1)	121(15.3)	84(7.6)

＊データが小さければ全字数は小さいのが普通であるから、字数の平均値にはほとんど意味はない．

それでもデータ圧縮においては漢字を短い符号語長で出力するため，漢字を使用しない圧縮法に比べて圧縮率は良くなっている．

4．2　実行時間の減少

LZSS 法および SQUEEZE 法では，漢字符号の使用によって圧縮処理に要する時間が短くなった．これは，表2に挙げたデータから次のように説明できる．漢字符号を使用する方法には，入力文字列を漢字符号化する過程と実際の圧縮処理を行う過程のふたつの過程がある．漢字符号が12ビットの場合は，この圧縮処理過程に必要な時間は，データの大きさではなく表2に挙げた文字数に比例することになる．すなわち，文字数比分だけ漢字を使用しない場合に比べて短くてすむ．また，漢字符号の語長が短くなれば圧縮過程で処理する文字数が多くなるので，実行時間はその分だけ多く要することになる．漢字符号化過程の実行時間は全ての方法にほぼ共通であるが，この圧縮過程の時間が全実行時間により大きく影響するため，LZSS 法および SQUEEZE 法では実行時間が減少することになる．

5．む　す　び

本論文では，日本語で書かれた文書を効率よく圧縮するための日本語の符号化法を提案した．提案した符号化法では，日本語文書における文字の出現頻度の解析結果から，頻度の多い順に文字を最大1024字選び，それらの日本語文字だけを漢字符号化し1文字として取り扱い，他の日本語文字は2バイトの2文字として取り扱う．その漢字符号語を使用して，LZW 法，LZSS 法および SQUEEZE 法の圧縮実験を行い，その日本語符号化法の有効性を実証した．その結果，圧縮率の面では漢字符号に選ぶ字数は1024字あるいは512字が最適の値であることを明らかにした．また，実行時間の面でも性能が改善されることも示した．

文　献

(1) 稲井義正：〝短い符号語長の漢字符号を使用した日本語文書のデータ圧縮〟，愛媛大学教育学部紀要，13，2，pp.57－63（1993－02）．
(2) Ziv J. and Lempel A. : "Compression of Individual Sequnces via Variable-Rate Coding", IEEE Trans. Inf. Theory, IT－24, 5, pp.530－536 (1978－09).
(3) Welch T. A. : "A Technique for High-Performance Data Compression", IEEE computer, 17, 6, pp.8－19 (1984－06).
(4) Storer J. A. and Szymanski T. G. : "Data Compression Via Textual Substitution", Journal of ACM, 29, 4, pp.928－951 (1982).
(5) Storer J. A. : "Data Compression : Methods and Theory", Computer Science Press, pp.323－334 (1988).