自 然 言 語 処 理 148－14
（２００２．３．５）

キーワードの境界推定のためのポテンシャル関数

田中 路子 † 梅村 恭司 †

† 豊橋技術科学大学 情報工学系
441-8580 豊橋市天伯町雲雀ヶ丘 1-1
TEL: 0532-47-0111(ext.5430)
michiko@ss.ics.tut.ac.jp, umemura@tutics.tut.ac.jp

日本語の処理を行う場合，英語とは異なり分かち書きを行う必要がある．このようにテキストを分割する際に用いられる手法として，ポテンシャル関数の合計の最大値を求める方法で単語の切り出しを行う手法がある．本稿では，このような手法を用いて，論文アブストラクトの筆者が付与したキーワードを切り出す問題におけるポテンシャル関数について報告する．そして実験より，その有効性の検証を行う．

A Potential Function to Detect Boundaries of Keywords

Michiko Tanaka† Kyoji Umemura†

† Dept. of Information and Computer Sciences, Toyohashi University of Technology
Tempaku, Toyohashi, Aichi, 441-8580, Japan,
TEL: (+81)532-47-0111(ext.5430)
michiko@ss.ics.tut.ac.jp, umemura@tutics.tut.ac.jp

Word segmentation is important for Japanese processing. It is possible to detect word boundaries that give maximum value of some potential function as total. This report proposes one function which is suitable to detect boundaries of keywords selected by authors. It also reports the evaluation results using these keywords.

1. はじめに

近年，様々な情報が電子化され，我々はこれらを利用することができる．しかし，日本語の処理を行う場合，まず，分かち書きを行う必要がある．一般にこのような処理は，形態素解析[1]を利用する．しかし，形態素解析のような辞書を用いた手法の場合，分割の精度が辞書に依存するという問題がある．すなわち，辞書に登録されていない未知語を処理できないため，常に辞書を最新の情報に更新する必要がある．しかし，このような処理は，日々新しい単語が生まれる現在の情報処理としては問題があると考える．

これに対し，辞書を利用せずにテキストから文字列を抽出する手法として，テキスト中の文字列の頻度情報によって，文字の切りだしを行うものがある．文献[2]では，情報検索に利用する索引語の切りだしを，tf・idf より求めたポテンシャル関数の合計の最大値を求める方法で行っている．これに対し，我々はこれまで $adaptation$[3]という出現集中を示す統計量を基にしたポテンシャル関数を用いたキーワード抽出手法を検討してきた[4]．しかし文献[4]では，キーワードらしい単語を取得できることについては確認したが，抽出した文字列のキーワードとしての正当性については評価していなかった．また，文献[4]では $adaptation$ をキーワードらしさの近似として用いたが，明らかにキーワードではない文字列に関しても，高い $adaptation$ の値を持つことが観測された．そこで本稿では，文献[4]の関数を置き換える新たなポテンシャル関数を定義すると共に，抽出した文字列を人間が選択したキーワードと比較することで評価を行う．なお本稿では，これ以後ポテンシャル関数を短くスコアと呼ぶこととする．

2. 文字列分割の原理

まず，文献[4]の文字列分割手法の原理を簡単に説明する．次に，文献[4]の文字列分割手法の問題点を述べ，この問題点を考慮した新たなスコアを定義する．

ここで，文字列分割手法の原理を述べるにあたり，必要な特徴量を定義する．

- $tf(x)$：ドキュメント集合全体に含まれる文字列 x の出現頻度
- $df(x)$：ドキュメント集合全体において文字列 x が出現したドキュメントの出現頻度
- $df_2(x)$：ドキュメント集合全体において文字列 x が 2 回以上出現したドキュメントの出現頻度
- $adaptation(x)$ ： 出現集中を示す特徴量で $df_2(x)/df(x)$ により求められる[3]．
- $tf(x)\cdot idf(x)$：一般に文書中の文字列の重要度を求めるために用いられる．なお，$idf(x)$ は $-\log(df(x)/N)$ により求められる．ここで，N はドキュメント数である．

これ以後，対象とする文字列 x が明らかな場合は，簡単のため引数 x を省略する．

2.1 従来の文字列分割手法の原理

文字列の単語らしさの指標となるスコアを，式 (1) に定義する．

$$score(x) = \begin{cases} -100 & df_2(x) \le 3 \\ \log(df_2(x)/df(x)) & df_2(x) > 3, df(x)/N \le 0.5 \\ \log(0.5) & df_2(x) > 3, df(x)/N > 0.5 \end{cases} \quad (1)$$

なお，式 (1) の各々の条件は次の考えによる．

$df_2(x) \le 3$ の条件：

キーワードではない文字列の存在範囲

$df_2(x) > 3, df(x)/N \le 0.5$ の条件：

キーワードの存在範囲

$df_2(x) > 3, df(x)/N > 0.5$ の条件：

高頻度な単語 (助詞など) の存在範囲

式 (1) において，単語の分割の指標となるスコアに $df_2/df(= adaptation)$ の対数を用いている．$adaptation$ は，キーワードに関して高い値を持つことが知られている特徴量である．

式 (1) のスコアを用いた文字列の分割は，このスコアの合計が最大になる分割を求めることで行う．図 1に文字列が 3 文字の場合の分割を示す．図 1の ABC の 3 文字に対する分割方法は 4 種類あり，一般に N 文字に対する分割は，2^{N-1} 種類となる．この中からスコアの合計が最大になる分割を求める手法として，ビタビ・アルゴリズム[5]を用いる．

2.2 従来の文字列分割手法の問題点

次に，文献[4]の分割手法では，正しく分割することのできない文字列の振る舞いの例を示す．ここでは，例として「ニューラルネットワーク・モデル」における文字列の分割を考える．図 2と

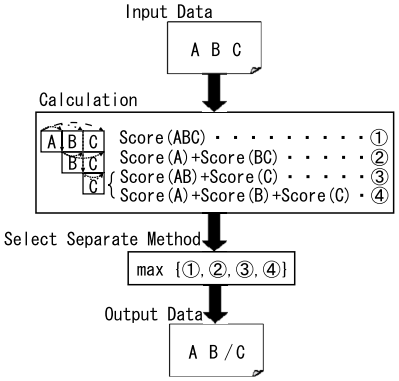

図 1　文字列の分割方法

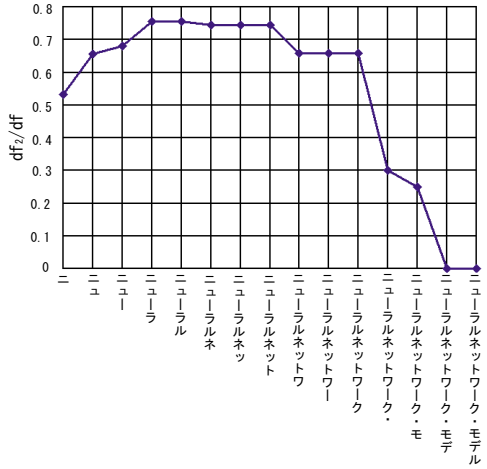

図 2　adaptation の変異

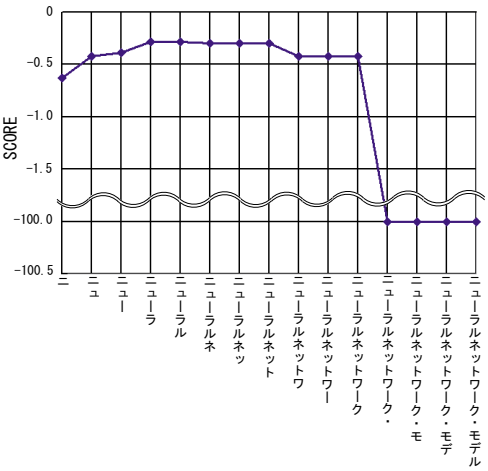

図 3　スコアの変異

図 3に，*adaptation* とスコアを示し，このスコアを用いた分割方法を図 4に示す．

「ニューラルネットワーク・モデル」という文字列の分割を考慮した場合，分割方法は，$2^{14}(=16384)$ 種類あり，この中からシステムが選んだ分割は，図 4より「ニューラルネットワー」「ク・モデル」であった．

ここで問題になるのは，スコアの値である．先に示したように，スコア付けは *adaptation* を用いて行っていた．つまり，スコアは文字列が出現したドキュメントの頻度によって決定される．しかし，単語の部分文字列は，分割したい単語と同等かそれ以上のドキュメントに出現するため，単語の部分文字列のスコアも単語のスコアと同程度の高い値を持つ．つまり，単語として成り立たない部分文字列によって，文字列の分割を行った場合でも，スコアの合計値は下がらない．部分文字列の *adaptation* やスコアが高い値を持つことは，図 2 ，図 3においても確認することができる．以上のことより，*adaptation* の値が下がる点にのみ着目していた従来の分割方法では，文字列の分割を行う指標として不十分だと考える．

2.3　文字列分割手法の改善法

本稿で目指す新たなスコアの特性は以下の二つである．

(1)　単語として成り立つ文字列の境界ではスコアは高いまま保持する．

(2)　確実に単語として成り立たない部分文字列による分割を防ぐため，そのような文字列のスコアを低く設定する．

我々は，単語として成り立たない文字列を発見するための指標として図 2の *adaptation* の変異に着目した．この図より，確実に単語として成り立たない部分文字列と言える「ニューラルネットワー」から,「ニューラルネットワーク」への *adaptation* の値が平坦なことが確認できる．

つまり，従来のキーワード抽出手法では，文字列を分割，抽出する統計量として *adaptation* の値が下がる点のみに着目していたのに対し，提案する改善法では，*adaptation* の値が下がる点に加え，*adaptation* の変異についても考慮した分割手法を検討する．なお，ここでは直接，*adaptation* の値を用いて文字列の分割を求める関数を定義するのではなく，*adaptation* が示す単語らしさが，さらに顕著に表れるよう改善を行った図 3に示すスコアを用いる．つまり，式 (1) で定義した *adaptation* の値が下がる点に着目したスコアの性質を利用しながら，新たなスコアの指標を定

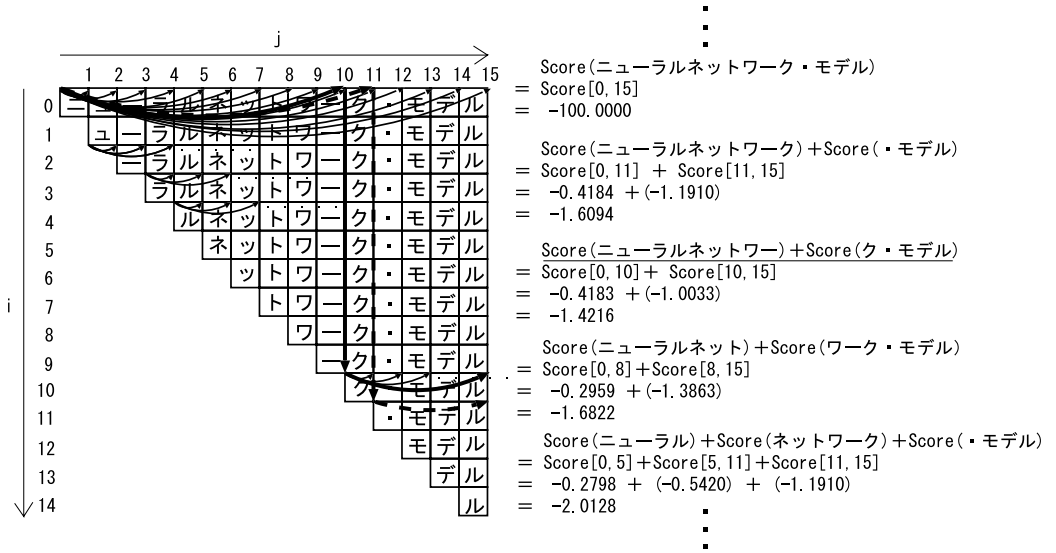

図 4　「ニューラルネットワーク・モデル」における文字列の分割

義することとした．そこで，式 (1) で定義したスコアをこれ以降では，old_score と呼び，本稿で定義する新たなスコアを new_score と呼ぶ．

新たなスコアの性質としては，図 3によっても確認できるように，対象とする文字列よりも長い文字列とのスコアの変異が小さいものに関しては，長い文字列の部分文字列と考え，スコアの値を低く設定するものである．そこで，考慮する文字列の終了位置を後ろへ進めた文字列とのスコアの変化と，文字列の開始位置を前へ進めた文字列とのスコアの変化を計算することで文字列のスコアの平坦さを求める．なお，ここではスコアの平坦さを求めることが目的であり，その符号は考慮しないため，変化量は絶対値で扱うこととする．これ以後，簡単のために，終了位置を後ろへ進めながら得たスコアの変化量を後変化分と呼び，開始位置を前へ進めながら得た文字列とのスコアの変化量を前変化分と呼ぶ．一例として，これまで検討してきた「ニューラルネットワーク・モデル」の文字列に対する，後変化分を図 5に示し，前変化分を図 6に示す．ここで，これらの図における X 軸は，変化量を求めた文字列を示しており，これらの文字列のスコアの変化分を Y 軸に示す．

図 5における「ニューラルネットワーク」と「ニューラルネットワーク・」との間の後変化分や，図 6における「ネットワーク・モデル」と「ルネットワーク・モデル」の前変化分が極端に大きい値を持っている．これは,「ニューラルネットワーク・」や「ルネットワーク・モデル」が語の境界を越え，式 (1) のキーワードではない文字列の領域へと文字列の存在範囲が変化したために，これらの文字列のスコアの値が極端に小さくなったためである．

「ニューラルネット」と「ニューラルネットワ」との間の後変化分や「モデル」と「・モデル」の間の前変化分のように，単語として成り立つものから，成り立たないものへと変化する際の $adaptation$ の変化は大きいが，それ以外の単独で単語として成り立たない部分文字列同士の変化は小さい．そのため，これらの文字列は今回の改善で着目しているスコアの平坦な部分に存在していることが確認できる．このことより，文字列の後変化分や前変化分を求めることで，単独で単語として成り立たない部分文字列を発見することができると考えた．

このことを踏まえ，単語として成り立たない部分文字列を発見するためのしきい値を導くため，「ニューラルネットワーク」のように，人間が見て単語と部分文字列が明確な文字列を 50 個選択し，それらの部分文字列の後変化分と前変化分を求めた．そして，各文字列のうち，分割された文字列が単語として成り立つものと成り立たないものを判断し，そのときの後変化分と前変化

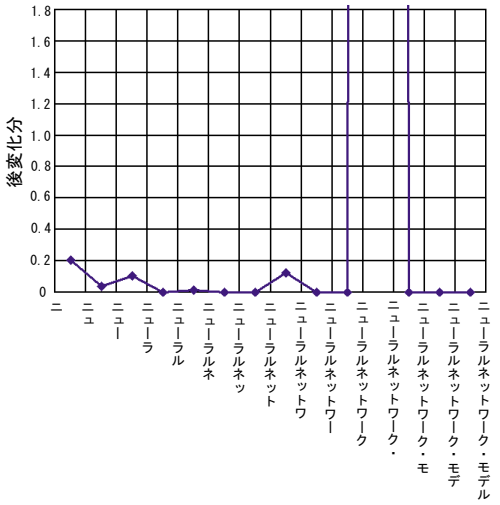

図 5　「ニューラルネットワーク・モデル」の後変化分

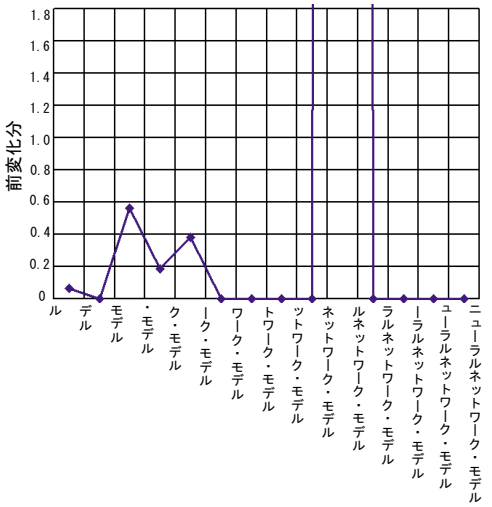

図 6　「ニューラルネットワーク・モデル」の前変化分

分の境界値をしきい値として用いる．また，これらの文字列に対し，文字列が分割されると成り立たない部分文字列の後変化分を重ね合わせた結果を図 7に示す．同様に，前変化分を重ね合わせた結果を図 8に示す．図 7，図 8の Flat と明記した部分が単独で単語として成り立たない部分文字列の変化量を示す．これらの図より，単独で単語として成り立たない部分文字列は，他の部分に比べて，特に小さい変化量を持っていることが確認できる．

図 7，図 8より，単独で単語として成り立つ部分文字列と，成り立たない部分文字列とのしきい値を 0.006 に決定した．すなわち，後変化分と前変化分が 0.006 以内なら，スコアを低く設定することとする．つまり，新しく定義するポテン

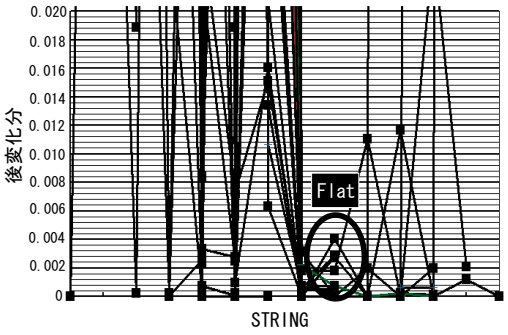

図 7　後変化分のしきい値の決定

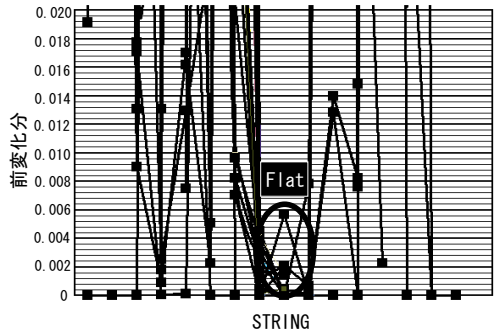

図 8　前変化分のしきい値の決定

シャル関数を次式のように定義する．

$$new\_score[i,j] = \begin{cases} old\_score[i,j] \\ \quad (|old\_score[i,j+1] - old\_score[i,j]| > 0.006 \\ \quad \wedge |old\_score[i-1,j] - old\_score[i,j]| > 0.006) \\ -100 \\ \quad (|old\_score[i,j+1] - old\_score[i,j]| \leq 0.006 \\ \quad \wedge |old\_score[i-1,j] - old\_score[i,j]| \leq 0.006) \end{cases} \quad (2)$$

上式において，i は文字列の開始位置を示し，j は文字列の終了位置を示す．なお，式 (2) において，単独で単語として成り立たない部分文字列と判断した場合のスコアを -100 としたが，これは式 (1) と同様に，この部分文字列を用いて分割を行わないという意味を込め，他のスコアに比べ極端に小さい値を設定した．

本稿で述べた改善策を含め，文字列の分割における処理の流れを図 9に示す．

以上のアルゴリズムによって，変更した「ニューラルネットワーク・モデル」のスコアを図 10に示す．図 10では，図 3と比較し，他の単語でも用いられるものか，単独で意味を持つ「ニューラルネット」や「ニューラルネットワーク」のような文字列以外のスコアが低く設定できていることが確認できる．

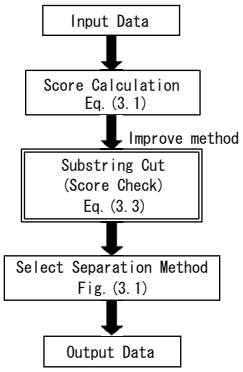

図 9　文字列除去の流れ

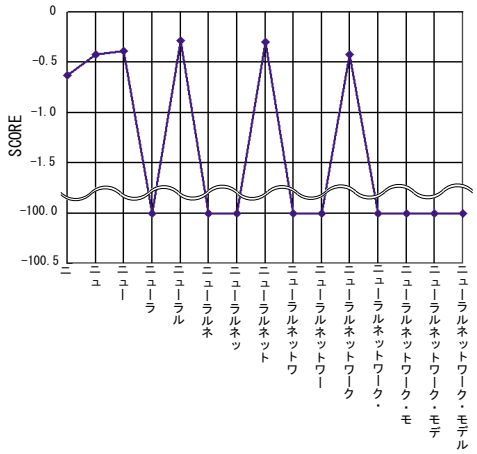

図 10　改善後のスコアの変異

3. 評　　価

ここでは，定義したポテンシャル関数を基に分割を行った文字列に対し，次式に示す条件を満たすものをキーワードとした．この条件は，従来法と提案法で同一である．

$$keyword = \begin{cases} score(\approx \log(df_2/df)) > -1.0 \\ 0.00005 < df/N < 0.1 \\ length > 1 \end{cases} \quad (3)$$

3.1　単語の切り出しによる評価

まず，システムがキーワードとして抽出した語がキーワードにふさわしいかということは判定せずに，単語として正しく切り出されているかという点についてのみ評価する．

ここで対象とするデータは，NTCIR の 33 万件の論文アブストラクト [6] における検索要求 30 件とする．表 1に文献 [4] によって報告された従来法と本稿で提案した改善法における，キーワードの切り出しによる評価を比較する．従来法では，システムが抽出した文字列が 202 個あり，そのうち，語として成り立っていたものが 175 個あった．そのため，語の切り出しの精度は 86.63[%] であったのに対し，改善後のシステムが抽出したキーワードの数は 195 個あり，このうち，語として成り立っているものが，従来のシステムと同じ 175 個であった．そのため，その精度は 89.74[%] へと向上した．ここで，注目すべきことは，改善法において誤った抽出が少なくなったことはもちろんのこと，システムが抽出した正解の数が減少していないことである．

次に，提案した改善が従来法の誤った抽出をどのように改善させたのかを表 2に示す．表 2より，提案した改善法によって抽出したキーワードは，文字列の意味ある固まりを捉えることができていることを確認できる．以上の結果より，文字列の切り出しにおいて，部分文字列を除去するという本改善法の有用性が確認できた．

次に，システムが抽出した文字列のキーワードとしての正当性と，本改善法による精度の変化について検討する．

3.2　キーワード抽出の精度による評価

評価において問題となるのは正解データであるが，ここでは図 11に示すような NTCIR の論文アブストラクト 33 万件に含まれる，論文の筆者が付与したキーワード (KYWD タグ内の文字列) を用いることとした．なお，本システムでは，この KYWD タグを除去したデータに対し，キーワード抽出を行った．但し，正解のデータとして用いる筆者が付与したキーワードには，KYWD のタグ内のみにしか出現しないキーワードがある．つまり，本システムが抽出を試みる KYWD タグ欄を除去した 33 万件のデータには，一度も出現しないキーワードが存在する．このようなキーワードは，筆者の主観や表記の揺れが存在すると考え，適切なキーワードではないと判断し，正解のキーワードから除去した．

ここでの評価尺度として，式 (4) に示す正解率と式 (5) に示す再現率を用いる．

$$\text{正解率} = \frac{\text{抽出したキーワードのうち正解の数}}{\text{抽出したキーワードの数}} \times 100 \quad (4)$$

$$\text{再現率} = \frac{\text{抽出したキーワードのうち正解の数}}{\text{正解のキーワード数}} \times 100 \quad (5)$$

さらにここでは，従来法との比較に加え，辞書を用いた手法との比較も行う．辞書を用いた手法は，キーワード抽出の最も単純な手法と考えら

表 1　文字の切り出しによる評価

	従来法	改善法
抽出したキーワードの数	202	195
切り出しに成功した数	175	175
切り出しに失敗した数	27	20
切り出しの精度	86.63[%]	89.74[%]

表 2　改善された文字列の例

単語の切れ目が正しく切れる	データ駆 $\Longrightarrow$ データ駆動
	ルゴリズム $\Longrightarrow$ アルゴリズム
	ニューラルネットワー $\Longrightarrow$ ニューラルネットワーク
誤った文字列を抽出しなくなった	,(
	ンコンピュータ
複合語としては捉えきれていないが意味のある固まりで分割している	人工ニュー + ラルネットワーク ⇓ 人工ニューラル + ネットワーク

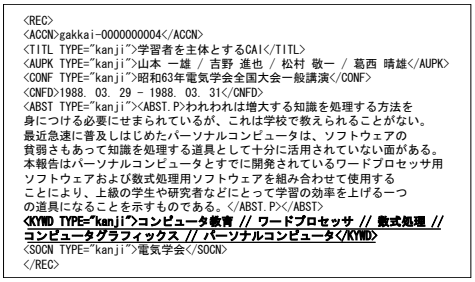

図 11　対象とする NTCIR コーパス

れる，形態素解析 (茶筌 version1.51) によって分割した文字列に対し，tf・idf を用いてキーワードの選別を行う手法を用いた．なお，茶筌の辞書については専門辞書の追加などは行わなかった．

実際にドキュメント集合から求めたキーワードの数と精度を表 3に示す．表 3より，従来法において，システムが抽出したキーワードは 91115 個あり，そのうち 25041 個が筆者が選択したキーワードと一致していたことより，システムの正解率は 27.48[%] であった．これに対し，改善法ではシステムが抽出したキーワードは 79792 個あり，そのうち 25026 個が筆者が選択したキーワードと一致していたことより，システムの正解率は 31.36[%] であった．ここでも，改善法によって抽出しなくなった単語の数が 11323 個と非常に多いにもかかわらず，正解であるものを捉えきれなくなった単語の数は，わずか 15 個であることからも，本改善法が，従来の正解を損なうことなく，精度向上が行える手法だということが確認できる．

以上より，改善法はキーワードの抽出においても精度向上できていることを確認できた．このことにより，*adaptation* の値が下がる点のみに着目していた従来法に比べ，*adaptation* の下がる点と，その変異の両方を考慮して文字列の分割を行うことの有効性を示すことができた．

これに対し，辞書を用いた手法は，再現率は提案法に比べ高いが正解率が低い精度になった．この最も大きな理由は，茶筌が複合語を捉えることが出来なかったことが考えられる．表 4に，茶筌が抽出できなかった複合語の例を示す．

次に，提案法の 31.36[%] という精度についてだが，この値はある意味で妥当な精度だと我々は考える．なぜなら，ここで正解データとして用いた論文アブストラクトの筆者が付与したキーワードは，多数の人間によって検討し導き出されたキーワードではなく，各々の筆者が，背景知識や主観をもって独断で付与したものであり，各個人ごとにキーワードの評価基準が異なるのみならず，その表記においても異なるものが存在する．そのため，本システムがキーワード抽出を行ったコーパスから，人間がキーワードの付与を行った場合でも，正解データである筆者の選択したキーワードと全く同じキーワードを抽出することは非常に困難だと考える．そのため，ここで示した評価は一つの指標にすぎないが，辞書も背景知識も持たない本システムが選択した 3 割のキーワードが，人間が選択したキーワードと完全に一致した文字列を捉えているということは，興味深い結果だと考える．

次に，システムが抽出したキーワードと，ドキュメント集合中の KYWD タグ欄への出現回数の関係を検討する．すなわち，33 万件の全ド

表 3　キーワードとしての評価

	従来法	改善法	辞書を用いた手法
抽出したキーワードの数	91115	79792	289306
抽出したキーワードのうち正解の数	25041	25026	38248
正解率 [%]	27.48	31.36	13.22
再現率 [%]	8.40	8.40	12.84

（ドキュメント集合中に含まれる正解のキーワード数：297849）

表 4　茶筌で抽出できなかった複合語

正解のキーワード	茶筌が分割したキーワード	正解のキーワード	茶筌が分割したキーワード
ロゴスキーコイル	ロゴ + スキー + コイル	知的障害者	知的 + 障害 + 者
発光ダイオード	発光 + ダイオード	阪神大震災	阪神 + 大震災
直列共振回路	直列 + 共振 + 回路	非難経路	非難 + 経路
PWM 制御	PWM + 制御	有害廃棄物	有害 + 廃棄物
フラッシュオーバ電圧	フラッシュオーバ + 電圧	古今和歌集	古今 + 和歌集
周波数応答	周波数 + 応答	特別養護老人ホーム	特別 + 養護 + 老人 + ホーム

表 5　キーワードタグ欄への出現回数と再現率の関係

ドキュメント数	正解数	改善法		辞書を用いた手法	
		抽出したうち正解の数	再現率 [%]	抽出したうち正解の数	再現率 [%]
1 個以上	297849	25026	8.40	38248	12.84
10 個以上	18766	9361	49.88	5747	30.62
100 個以上	1396	938	67.19	663	47.49
200 個以上	499	353	70.74	268	53.71

キュメントにおける KYWD タグ欄への出現回数によって正解データの分類を行い，それらの正解データを，システムがどれだけもらさずに捉えることができているかという再現率を求めることで評価を行う．なお，従来法と改善法は，文書においてキーワードとして捉えている部分は一致しているため，再現率は同程度の精度となる．このことは，表 3の再現率が同じ値を示していることからも確認できる．そのため，ここでは改善法についてのみ検討を行うこととする．

ドキュメント中の KYWD タグ欄へ出現したキーワードの数 (正解数) と，システムが抽出したキーワードのうち正解のキーワードと一致した数と再現率を表 5に示す．表 5より，多くのドキュメントに出現したキーワードほど，システムが抽出したキーワードの再現率が高くなっていることが確認できる．また，この表より $tf \cdot idf$ によって選別を行った辞書を用いた手法に比べ，提案法の方が，この特徴が顕著に表れていることが確認できる．以上のことより，本システムが抽出された文字列は，キーワードとして正しい単語を取得できていると考える．

4. ま と め

本稿では，テキスト中のキーワードを切り出す問題に対し，*adaptation* という出現集中を示す統計量が下がる点に加え，統計量の変異に注目したポテンシャル関数を提案した．実験により，論文アブストラクトの著者が選択したキーワードと評価し，実際に効果のあることを確認した．

謝辞　この研究は平成 12 年度 IPA 未踏ソフトウェア創造事業のプロジェクトの一部として実施した成果を利用させていただきました．深く感謝いたします．

参 考 文 献

1) Yuji Matsumoto, Akira Kitaushi, Tatsuo Yamashita, Yoshitaka Hirano, Osamu Imaichi and Tomoaki Imamura.: Japanese morphological analysis system chasen manual, NAIST,Technical Report NAISTIS-TR97007, (1997).
2) 小澤智裕，山本幹雄，山本英子，梅村恭司：情報検索の類似尺度を用いた検索要求文の単語分割，言語処理学会大会，A5-2, (1999).
3) Kenneth W. Church.: Empirical Estimates of Adaptation, Coling-2000, pp. 180-186, (2000).
4) 武田善行，梅村恭司：キーワード抽出を実現する文書頻度分析，計量国語学 vol.23，No.2，(2001).
5) 北 研二: 確率的言語モデル, pp. 113, 東京大学出版会.
6) Noriko Kando, Kazuko Kuriyama, Toshihiko Nozue, Koji Eguchi, Hiroyuki Kato and Souichiro Hidaka.: Overview of IR Tasks at the First NTCIR Workshop, NTCIR Workshop, vol. 1, pp. 11-44, (1999).