Vol.2012-SLP-94 No.15
2012/12/21

複数認識結果を用いて構築した Suffix Array に対する音声検索語検出

勝浦広大[†1]　桂田浩一[†1]　入部百合絵[†1]　新田恒雄[†1†2]

本論文では suffix array を用いた高速で精度の高い音声検索語検出法を提案する．これまで筆者らは，単一の音節言語モデルを用いて認識した結果から suffix array を構築していたが，本研究では，単語と音節の言語モデルを用いて音声認識を行い，出力された複数の認識結果を統合し suffix array を構築することで検索性能の向上を実現する．評価実験の結果，CSJ データベースに対して CORE 講演 44 時間分では，F 値の最大値が約 9 ポイント向上し，MAP は約 7 ポイント向上した．また，ALL 講演 604 時間分では，F 値の最大値が約 12 ポイント向上し，MAP は約 9 ポイント向上した．さらに，複数の認識結果から構築した suffix array では再現率あたりの検索時間が，単一の認識結果から構築したものと比べて短いことが分かった．この結果より，複数認識結果を用いることで高速で性能のよい音声検索語検出を実現できることが分かった．

Spoken Term Detection for Suffix Array was Constructed Using Multiple Speech Recognition Results

KOUDAI KATSUURA[†1]　KOUICHI KATSURADA[†1]
YURIE IRIBE[†1]　TSUNEO NITTA[†1†2]

This paper proposes a fast spoken term detection (STD) method using a suffix array. Previously, we have proposed a STD method in which a suffix array is constructed from a recognition result obtained by using a single syllable-based language model. In this study, we attempt to improve search performance by constructing the suffix array from two or more recognition results obtained by using word-based and syllable-based language models. Experimental results show that the maximum F-measure and MAP scores increased by 9 points and 7 points on the CSJ core lectures (44 hours), respectively, and those on the CSJ all lectures (604 hours) increased by 12 points and 9 points, respectively. Moreover, we confirmed the search time per the recall rate decreased by using the suffix array constructed from multiple recognition results. These results show that constructing suffix array from multiple recognition results makes the search performance considerably improve without increasing the search time severely.

1. はじめに

近年 web 上での音声データや動画像データの需要が高まり，そのコンテンツ数も大幅に増加しつつある． こうしたデータを有効活用する一つの方法が音声検索語検出である．音声検索語検出とは与えられたキーワードの出現箇所を音声データ内から検索する技術を指す．音声検索語検出に関する研究は近年盛んに行われており[1]，2006 年に NIST の主催でベンチマークテストが行われた[2]のを始めとして，日本においても NTCIR-9 のタスクに組み込まれるなど[3]，共通のタスクによる客観的な評価が行われ始めている．特にここ数年の間に，数十～数千時間の音声データを対象に数ミリ秒～数秒で結果を出力する非常に高速な検索法が提案されており，音声検索語検出における重要な課題の一つとなっている[4][5][6]. 筆者らも suffix array に DP マッチングを適用することによって，非常に大規模な音声ドキュメントから高速にキーワードを検出する手法を提案している[7][8]．本研究では，複数認識結果を用いることにより，高速性を保ちつつ検索性能の向上を実現する手法を検討する．

これまでに複数の認識結果を使用する手法はいくつか提案されている．最も一般的な手法は，音素 N-gram 言語モデルと単語 N-gram 言語モデルの二つの認識結果を使用した方法である[6][16]．この方法では，キーワードが与えられたときに，キーワードの種類に応じて二通りの検索処理を適用する．与えられたキーワードが既知語の場合，単語 N-gram 言語モデルを用いて得られた認識結果を検索対象とする．また，キーワードが未知語の場合は，音節 N-gram 言語モデルを用いて得られた認識結果を検索対象とする．キーワードが既知語の場合は言語モデルのバイアスが働くため，検索性能が大幅に向上する．

一方，他にもいくつか方法が提案されており，例えば西崎らは 5 種類の言語モデルと 2 種類の音響モデルから 10 種類の音声認識結果を出力し，得られた音声認識結果からコンフュージョンネットワークを構築し，その上で DP マッチングを適用することによって，20%以上の検索性能の改善を達成している[9]．この手法の利点は，様々な種類の認識結果を用いて検索性能を向上させるため，十分な拡張性を持っているということである．これはより精度の高い音声検索語検出のために望ましい特徴であると言える．しかし，この手法はコンフュージョンネットワークの始端から終端まで DP マッチングを適用するため，検索速度は高

†1 豊橋技術科学大学
Toyohashi University of Technology
†2 早稲田大学
Waseda University

Characters	Index	Suffix
a	1	abracadabra
b	2	bracadabra
r	3	racadabra
a	4	acadabra
c	5	cadabra
a	6	adabra
d	7	dabra
a	8	abra
b	9	bra
r	10	ra
a	11	a

Sort

Index	Suffix
11	a
8	abra
1	abracadabra
4	acadabra
6	adabra
9	bra
2	bracadabra
5	cadabra
7	dabra
10	ra
3	racadabra

Array={1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11}　Suffix Array={11, 8, 1, 4, 6, 9, 2, 5, 7, 10, 3}

図 1　Suffix Array の構築方法
Figure 1　How to construct a suffix array.

	a	i	u	e	o	k	s	t	…
高舌性	-	+	+	-	-	+	-	-	
低舌性	+	-	-	-	-	-	-	-	
前方性	-	-	-	-	-	-	+	+	
後方性	+	-	+	-	+	+	-	-	
破裂性	-	-	-	-	-	+	-	+	
:									

図 2　弁別特徴テーブルの一部
Figure 2　A fragment of a distinctive phonetic feature table.

速でないという問題がある．

本論文では，複数認識結果から suffix array を構築することにより高速かつ精度の高い音声検索語検出法を提案する．suffix array は複数の認識結果を音素列に変換したデータと，音素列を疑似木構造として格納するインデックスデータから構成される．複数認識結果を用いる場合，音素列 A の後ろに音素列 B を追加する形で保持し，インデックスデータは統合された音素列に対して作成される．検索時には正解箇所が複数得られる場合があるため，重複削除処理を行っている．

以下，第 2 節では，Suffix Array を用いた高速な音声検索語検出法の概要について述べる．第 3 節では，実験の概要と評価実験の結果について示し，最後に第 4 節で，本論文のまとめと今後の課題について述べる．

2. Suffix Array を用いた高速な音声検索語検出

2.1 手法の概要

Suffix array（接尾辞配列）[10]は，テキスト中の全ての音素に対する index を格納した配列を，suffix（接尾辞）の辞書順にソートしたものである．図 1 に suffix array の構築例を示す. Suffix array はソート済みのデータ構造であるため，検索キーワードを効率的に見つけ出すことができるが，オリジナルの suffix array では完全一致検索を想定している．

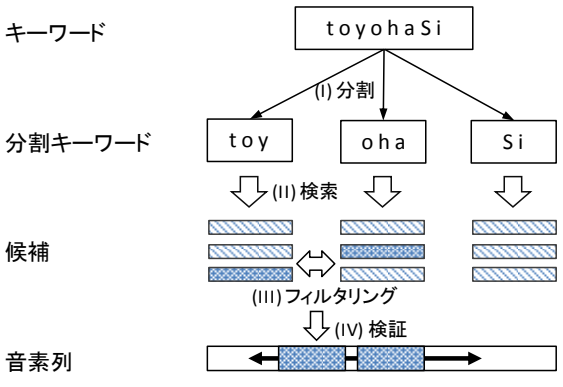

図 3　キーワード検索の流れ
Figure 3　The flow of keyword search.

このため，誤認識を含む音声認識結果を対象とする場合には何らかの曖昧検索技術を導入する必要がある．そこで本手法では山下らによって提案された，suffix array に DP マッチングを適用する方法[11]を導入して音声の誤認識に対応している．DP マッチングの局所距離には音素弁別特徴[12]を利用している．音素弁別特徴とは調音様式・調音位置によって音素を特徴付けしたもので，図 2 に示すように+またはーを取る 15 次元の素性により音素が表される．各音素間でこの素性のハミング距離を求め，局所音素間距離としている．

2.2 キーワード分割

Suffix array 上で DP マッチングを用いてキーワードを検索する場合，DP マッチングの実行中に枝刈りの閾値内のすべてのパスが保持されるため，閾値が大きいと探索空間および処理時間が指数関数的に増加することが，山下らによって確認されている．閾値は検索キーワードの長さに比例して増加させる必要があるため，検索キーワード長に対して指数的に処理時間が増大する．そこで，この問題を解決するためにキーワードを分割し，分割キーワードを元のキーワードの代わりに検索する手法を導入する．

本手法ではキーワードを分割した場合に分割しない場合と同一の検索結果が得られるよう，図 3 に示す 4 ステップからなる検索を行う．検索は (I)分割，(II)検索，(III)フィルタリング，(IV)検証の各ステップから構成される．

(II)検索のステップでは，式(1)の T_s のように分割キーワードの閾値を設定する．

$$T_s = \frac{T}{n-m+1} \tag{1}$$

ここで T_s は分割キーワードの閾値，T はキーワード全体の閾値，n はキーワードの分割数，m は検出されるべき分割キーワードの数である．検索の詳細および式(1)の導出過程については文献[13]を参照されたい．

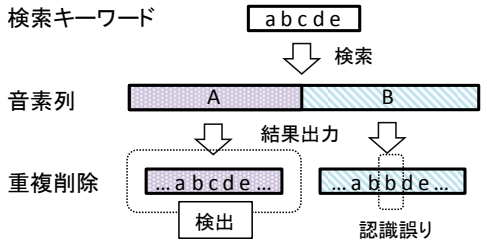

図 4　複数認識結果を用いた検索の流れ

Figure 4　The flow of search by multiple recognition results.

2.3 複数認識結果を用いた Suffix Array の検索手法

筆者らはこれまでに単一の認識結果を用いた検索手法を提案してきた[7][8]．これまでの検索手法に対して，高速性を保ちつつ精度の向上を図るために，複数認識結果から suffix array を構築し検索する手法を提案する．図 4 に提案手法の流れを示す．suffix array は対象音声を音素列に変換したデータと音素列を疑似木構造として格納するインデックスデータから構成される．複数認識結果を用いる場合，音素列は音素列 A の後ろに音素列 B を追加する形で保持し，インデックスデータは統合された音素列に対して作成される．本研究では，従来研究において効果的であることが確かめられている，単語言語モデルと音節言語モデルを用いて認識した結果を利用する．本手法では一つの検索キーワードに対して A と B の双方から検索結果が得られる場合がある．そこで一旦検索結果が得られた後に，正解箇所の重複削除処理を行っている．重複削除処理では，最もキーワードとの距離が近い結果のみが最終的な検索結果として残される．

3. 評価実験

3.1 実験環境と実験の概要

実験は Intel Core i7-2600 プロセッサ 3.4GHz，メインメモリ 8GB を搭載した PC で行った．実験で用いた音声データは NTCIR-9 の STD ワーキンググループにより提供された word-based transcription と syllable-based transcription（CORE 講演，ALL 講演）を用いた．これは CSJ コーパスをそれぞれ単語言語モデルと音節言語モデルで認識したものであり，CORE 講演が 44 時間分，ALL 講演が 604 時間分の音声データを含む．本手法では単語，音節を音素に変換し，suffix array に格納したものを対象に検索を行った．

本論文では予備実験の結果[7][8]から式(1)の m の値を 1 と設定し，分割キーワードの音素数が 6 音素になるよう分割することにした[a]．また，脱落，挿入ペナルティは 3.0 と設定した．さらに，短いキーワードの結果出力を抑えるために，キーワードの長さ l を考慮に入れた式(2)の *score* を用いて検索を実施した．t はキーワード一音素あたりの閾値，$t=T/l$ である．

$$score = \frac{1}{t / l^{1/2} + 1} \tag{2}$$

3.2 検索性能と速度の評価

表 1 と表 2 に CORE 講演，ALL 講演を対象としたときの本手法の score と再現率，適合率，F 値および検索時間を，図 5 と図 6 に NTCIR-9 で示されたベースラインの検索性能，従来手法である単一認識結果を用いた性能，および本手法の性能を再現率－適合率曲線で示す[3][14][b].

まず検索性能については，表 1～表 3 および図 5，図 6 に示すように，本手法は F 値の最大値，Mean Average Precision（MAP，平均適合率の平均）ともにベースライン

表 1　CORE 講演の検索性能

Table 1　Search performance of CORE-lecture experiment.

score	再現率(%)	適合率(%)	F 値	検索時間[ms]
1.000	39.9	90.6	55.4	0.3
0.970	41.0	90.9	56.5	0.3
0.940	42.7	91.2	58.2	0.0
0.910	48.8	89.4	63.1	1.24
0.886	**55.3**	**84.9**	**67.0**	**0.92**
0.880	58.1	68.0	62.7	0.94
0.850	66.1	60.6	63.2	3.14
0.820	71.6	25.8	38.0	9.04
0.790	77.1	16.8	27.5	31.5
0.760	80.4	6.96	12.8	117
0.730	85.1	1.37	2.69	290

表 2　ALL 講演の検索性能

Table 2　Search performance of ALL-lecture experiment.

score	再現率(%)	適合率(%)	F 値	検索時間[ms]
1.000	40.5	83.9	54.6	0.94
0.970	41.7	84.3	55.8	0.92
0.940	43.5	84.0	57.3	1.26
0.920	**49.3**	**74.3**	**59.2**	**2.8**
0.910	50.7	67.2	57.8	2.8
0.880	55.4	34.0	42.1	5.62
0.850	60.3	27.2	37.5	20.9
0.820	66.5	11.8	20.1	63.4
0.790	71.4	3.99	7.56	243
0.760	75.8	1.31	2.58	1016
0.730	81.9	0.38	0.76	2855

a) 分割で生じる余りの音素は分割キーワードに含めないことにした．余りの音素を分割キーワードに含めなくとも，式(1)に従えばキーワード全体の累積距離が T 以内のものを不足なく検索できる．

b) NTCIR-9 での筆者らの報告[3][14]では ALL の検索対象データに CORE のデータを含めないという誤りがあったため，文献[3][14]と結果が大幅に異なっている．また，CORE についても各キーワードの検索数の上限をなくしたため，性能に若干の違いが出ている．

表 3　NTCIR-9 参加グループの検索性能と検索時間

Table 3　Search performance and process time achieved by NTCIR-9 participants.

		F 値 最大値	Mean Average Precision	検索時間 [ms] (F 値最大)
CORE-lecture Experiment	ベースライン	52.7	59.5	36,400
	単一認識結果	58.0	66.1	1.54
	複数認識結果	67.0	73.2	0.94
	Nishizaki ら	72.5	83.7	13,440
	Kaneko ら	38.5	27.2	1.3
	Iwami ら	64.5	49.1	1.6
ALL-lecture Experiment	ベースライン	45.9	45.1	548,000
	単一認識結果	47.5	50.6	1.54
	複数認識結果	59.2	59.4	2.8

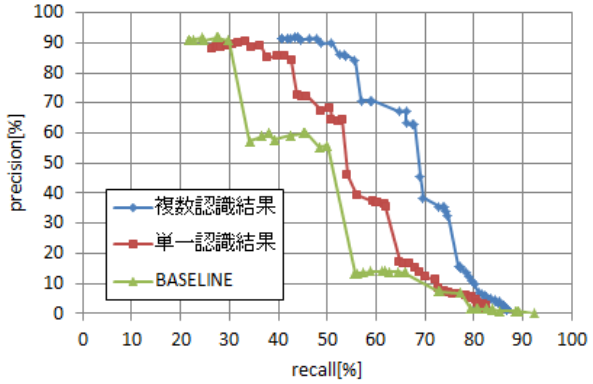

図 5　CORE 講演の再現率－適合率曲線

Figure 5　Precision-recall curve of CORE-lecture experiment.

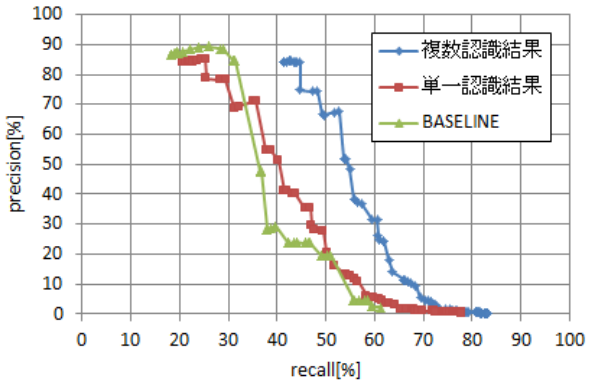

図 6　ALL 講演の再現率－適合率曲線

Figure 6　Precision-recall curve of ALL-lecture experiment.

を上回る性能を得ている．また，単一認識結果を用いた手法よりも性能が高いことが確認できた．しかし，Nishizaki らの手法[9]と比較すると，検索性能が十分とは言えないことが確認できる．

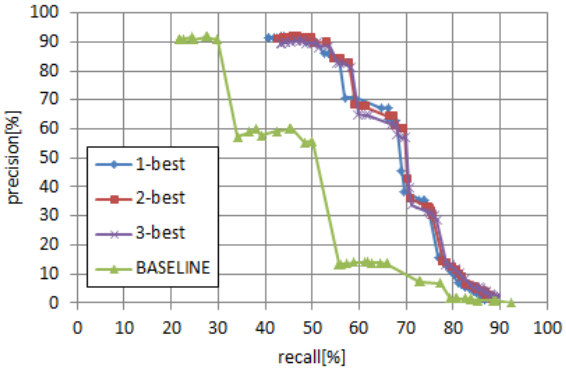

図 7　CORE 講演(3-best)の再現率－適合率曲線

Figure 7　Precision-recall curve of CORE-lecture experiment (3-best).

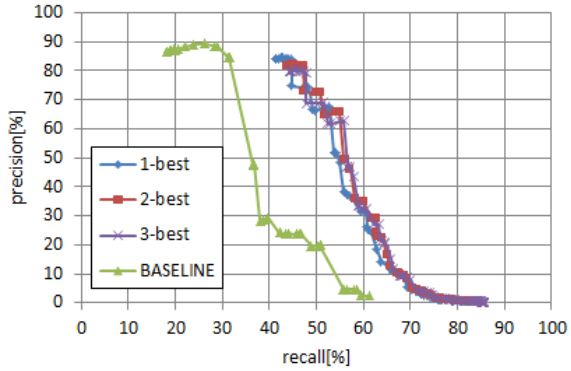

図 8　ALL 講演(3-best)の再現率－適合率曲線

Figure 8　Precision-recall curve of ALL-lecture experiment (3-best).

図 5，図 6 では，NTCIR-9 の STD ワーキンググループから提供された word-based transcription と syllable-based transcription のそれぞれ 1-best のみを使用し suffix array を構築している．これに対して，図 7，図 8 では 3-best までの単語と音節の結果を用いて suffix array を構築している．図 7 と図 8 の結果から，1-best，2-best，3-best それぞれで結果の違いはほとんど見られないことが分かる．また，図 9 に ALL 講演において，どの認識結果が最も効果的かを調査するために行った実験の結果を示す．単語と音節の 1-best から得られる認識結果を基本とし，その suffix array に単語と音節の 5-best までの結果を一つずつ統合し検索を行った．こちらも結果にほとんど差が見られなかった．これは，提供された word-based transcription と syllable-based transcription の 1-best から 5-best までがほぼ同様の認識結果を出力しているためである．本手法では DP マッチングによって認識結果から距離の近い範囲は全て検索対象になる．1-best から 5-best までが近い音素列の場合，本手法では，検索性能はほとんど変わらずインデックスデータの容量の

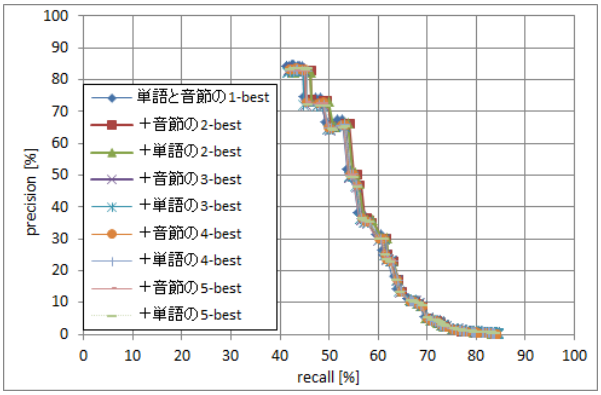

図 9　ALL 講演(5-best)の再現率－適合率曲線

Figure 9　Precision-recall curve of ALL-lecture experiment (5-best).

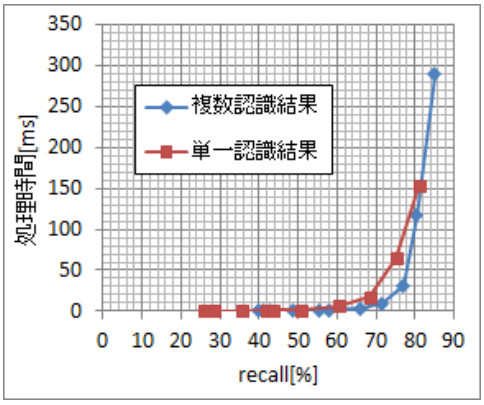

図 10　CORE 講演の検索速度

Figure 10　Search time of CORE-lecture experiment..

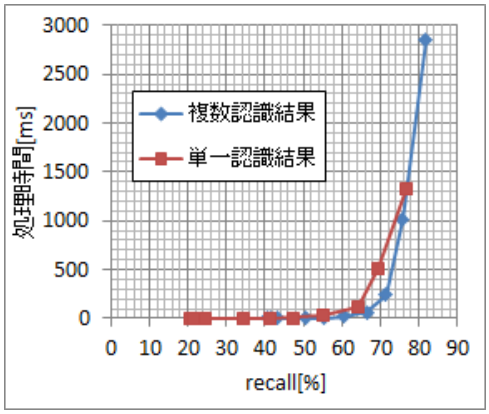

図 11　ALL 講演の検索速度

Figure 11　Search time of ALL-lecture experiment.

みが増えるというデメリットが生じる．Nishizaki らは異なる言語モデルと音響モデルを用いて作成した認識結果をもとに形成したコンフュージョンネットワークを用いることで，検索性能を大幅に向上させることに成功している．本手法においても同様に異なる言語モデル，音響モデルを用いて大きく異なる複数の認識結果を取り入れることで性能が向上するものと期待できる．

検索速度については，図 10，図 11 に CORE 講演と ALL 講演の複数認識結果と単一認識結果を用いた場合の結果を示す．再現率 60%～80%付近では，単一認識結果に比べて複数認識結果の方がより高速に結果を出力できていることが分かる．これは認識結果を増やしたことにより，正解の検出数が増えたためである．また，表 1 および表 2 に示すように，本手法は適合率が高い結果を高速に出力できていることが分かる．表 3 に示すように， F 値が最大になる検索結果を出力するのに要する時間は CORE 講演で 0.94ms, ALL 講演で 2.8ms となっている．計算機環境が異なるため単純な比較はできないが，本手法は NTCIR-9 参加グループの中でも Kaneko らの手法[15]，Iwami らの手法[16]と並んで最速グループに入る．ベースラインの連続 DP マッチングを用いる手法は ALL 講演の検索に 548 秒を要しており，suffix array の導入により大幅な高速化を実現できていることが分かる．

4. おわりに

本論文では複数認識結果を用いて構築した suffix array に対する音声検索語検出法を提案した．実験の結果，高速性を保ちつつ高精度な検索を実現でき，suffix array を用いた音声検索語検出において複数認識結果を用いる方法が有効であることが確認できた．しかし，提供された認識結果の 3-best までを利用した場合，精度の向上は見られたかったため，より精度の向上を図るためには 1-best とは大きく異なる新たな認識結果を追加することが必要である．

本手法は適合率が高く信頼性が高い検索結果を高速に出力できるという利点がある一方，再現率を高くするために閾値を緩めると検索候補，検索時間が指数的に増加するという問題がある．今後は検索時間の指数的増加を抑えるためのキーワード分割方法の最適化，他手法との組み合わせについて検討すると共に，新たな認識結果を得るために言語モデル，音響モデルを作成し，さらなる精度向上を目指したい．

参考文献

1)　秋葉友良：音声ドキュメント検索の現状と課題，情報処理学会研究報告，Vol.2010-SLP-82，No.10，pp.1-8(2010).
2)　Fiscus, J. G., Ajot, J., Garofolo, S. H. and Doddington, G.: Results of the 2006 spoken term detection evaluation, Proc. SIGIR'07 Workshop, pp.51-57 (2007).
3)　Akiba, T., Nishizaki, H., Aikawa, K., Kawahara, T. and Matsui, T.: Overview of the IR for Spoken Documents Task in NTCIR-9 Workshop, Proc. NTCIR-9 Workshop Meeting, pp.223-235 (2011).
4)　Pinto, J., Szoke, I, Prasanna, S. R. M. and Hermansky, H.: Fast approximate spoken term detection from sequence of phonemes, Proc. SIGIR'08 Workshop, pp.28-33 (2008).
5)　Wallace, R., Vogt, R. and Sridharan, S.: Spoken term detection using fast phonetic decoding, Proc. ICASSP2009, pp.2135-2138 (2009).
6)　Kanda, N., Sagawa, H., Sumiyoshi, T. and Obuchi, Y.: Open-vocabulary keyword detection from super-large scale speech

database, Proc. 2008 IEEE Workshop on Multimedia Signal Processing, pp.939-944 (2008).
7) Katsurada, K., Teshima, S. and Nitta, T.: Fast keyword detection using suffix array, Proc. InterSpeech2009, pp.2147-2150 (2009).
8) Katsurada, K., Sawada, S., Teshima, S., Iribe Y. and Nitta, T.: Evaluation of Fast Spoken Term Detection Using a Suffix Array, Proc. InterSpeech2011, pp.909-912 (2011).
9) Nishizaki, H., Furuya, H., Natori, S. and Sekiguchi, Y.: Spoken Term Detection Using Multiple Speech Recognizers' Outputs at NTCIR-9 SpokenDoc STD subtask, Proc. NTCIR-9 Workshop Meeting, pp.236-241 (2011).
10) Manber, U. and Myers, G.: Suffix arrays: a new method for on-line string searches, SIAM Journal on Computing, Vol.22, No.5, pp.935-948 (1993).
11) 山下達雄，松本祐治：Suffix Array を用いたフルテキスト類似用例検索，情報処理学会研究報告，Vol.1997-NL-97，No.85，pp.83-90 (1997).
12) Fukuda, T. and Nitta, T.: Orthogonalized distinctive phonetic feature extraction for noise-robust automatic speech recognition, IEICE Trans., Vol.E87-D, No.5, pp.1110-1118 (2004).
13) 桂田浩一，入部百合絵，新田恒雄：Suffix Array を用いた高速 STD におけるキーワード分割に関する理論的検討，情報処理学会研究報告，Vol.2011-SLP-89，No.16，pp.1-6 (2011).
14) Katsurada, K., Katsuura, K., Iribe, Y. and Nitta, T.: Utilization of Suffix Array for Quick STD and Its Evaluation on the NTCIR-9 SpokenDoc Task, Proc. NTCIR-9 Workshop Meeting, pp.271-274 (2011).
15) Kaneko, T., Takigami, T. and Akiba, T.: STD based on Hough Transform and SDR using STD results: Experiments at NTCIR-9 SpokenDoc, Proc. NTCIR-9 Workshop Meeting, pp.264-270 (2011).
16) Iwami, K. and Nakagawa, S.: High speed spoken term detection by combination of n-gram array of a syllable lattice and LVCSR result for NTCIR-SpokenDoc, Proc. NTCIR-9 Workshop Meeting, pp.242-248 (2011).