

音声検索語検出のためのクエリ拡張の検討

南條 浩輝^{1,a)} 前田 翔¹ 吉見 毅彦¹

概要：音声中で検索語(クエリ)がそのまま現れる部分を特定する音声検索語検出(Spoken Term Detection: STD)の研究を行う。STDでは、検索対象に音声認識誤りが存在するために、検索語ではないものも検出する誤検出問題が発生する。本研究では、この誤検出をできるだけ少なくする方法を検討する。具体的には、クエリ拡張を行い、拡張語での検索結果を参照することで、検索語の誤検出を抑制する方法について研究を行う。本論文では、拡張語の獲得方法として、クエリの関連語を拡張語とする手法とクエリの前後に文字列を付加したものを拡張語とする手法とを提案する。特に、後者の方法は全ての検索語に対して容易に拡張語を自動生成できるため、汎用性が大きいと考えられる。講演音声を対象としたSTDを行ったところ、関連語を拡張語とする手法およびクエリの前後に文字列(格助詞)を付加して拡張語とする方法ともに効果があることがわかった。後者の方法を種々のSTD検索タスクで評価したところ、全てのタスクで検索精度の大きな向上が得られ、提案手法の有効性および汎用性を示した。

1. はじめに

デジタル化された大量の音声や動画(音声ドキュメント)から、ユーザが知りたい特定の区間を取り出す音声ドキュメント検索[1][2][3]技術が求められている。音声ドキュメント検索では、音声認識によってテキスト化したものを対象に検索するのが一般的である。ただし、音声認識における音声認識誤りは本質的に避けられず、何らかの対処が必要である。これに対して、音声中の検索語(クエリ)の位置を見つける音声検索語検出(Spoken Term Detection: STD)[4][5][6][7][8]が注目されている。

音声認識が完全でない状況で、音声中の検索語の位置を漏れなくみつめるには、単語単位での検索は適さない。このため、単語よりも小さい単位(サブワード)で、誤りを考慮して照合を行うなどの工夫が必要である。その際、検索語ではないものを検出する誤検出問題が発生する。また、このようなSTDでは、音声認識誤りに由来する問題以外の問題も発生する。1つ目は同音異義語すなわち同じサブワード列(例えば音素列)でも意味が異なる単語が存在する問題である。例えば、アメリカ大統領の「オバマ」が出現するファイルを検索する場合に、「小浜(地名)」や「小濱(人名)」なども検出するという問題である。2つ目は文字列の部分一致の問題である。これは特にクエリ長が短いクエリで問題となると考えられる。例えば、「鯛 /t ai/」と

いうクエリで検索すると、「体育 /t ai i ku/」「ネクタイ /ne ku tai/」などを含む音声も検出するという問題である。

本研究では、これらの誤検出をできるだけ少なくする方法を検討する。具体的には、クエリ拡張を行い、拡張語での検索結果を参照することで、検索語の誤検出を抑制する方法について研究を行う。

2. 音声検索語検出

2.1 概要

音声検索語検出(Spoken Term Detection: STD)とは、音声中で検索語(クエリ)がそのまま現れる部分を特定する処理のことである。

STDの方法として様々な研究がなされているが、最も一般的かつ広く利用されているのは、音声を音声認識システムにより音声認識結果、すなわち文字列に変換しておき、これとクエリ文字列との間で文字列マッチングを行う方法である。

2.2 連続DPマッチングによる音声検索語検出

音声認識結果とクエリとの間のマッチングには、連続DPマッチングを用いることができる。クエリは一般的に単語列であるため、音声認識誤りがあると単語単位のマッチングはうまく機能しない。なお、音声認識誤りは本質的に避けられないため、この問題には何らかの対処が必要不可欠である。

このような背景に基づき、STDでは単語より小さな単位

¹ 龍谷大学理工学部
Faculty of Science and Technology, Ryukoku University
^{a)} nanjo@rins.ryukoku.ac.jp

であるサブワード（音素など）の単位でサブワード系列どうしを誤りを許容して照合（連続 DP マッチング距離が近いものを検索結果として選ぶ）することが一般的である。例えば、音声認識の際に *o: s a k a* という語に音声認識誤りが起こり、*o s a k o* として認識結果に登録された場合、完全一致ではそれを検索できない。そこで、サブワード間の編集距離が 2 までのものを検索することで検出が可能になる。しかしこの時、意図しないものも検出してしまう問題が起こる。例えば、*o: s a k i* や *o: s u g a* が出現する音声などを検出する。

本研究で用いる連続 DP マッチングに基づく STD のアルゴリズムは次のとおりである。

連続 DP マッチングに基づく STD

- (1) ユーザーからクエリ（長さ L ）を受け取る
- (2) 各音声ファイルを検索（STD）し、検索結果（音声ファイル名とそのファイル中の何番目の発話かの情報）を出力する。その際、クエリとの距離の近さを表すスコア $(1 - \frac{\text{Edit Distance}}{L})$ を付与し、その順に出力する。

3. クエリ拡張

情報検索におけるクエリ拡張とは、ユーザが入力した初期クエリに対していくつかの拡張語を加えることでより検索に適したクエリを生成する手法である。本研究では、STD においてクエリ拡張を検討するが、本論文でのクエリ拡張とは、新たなクエリを生成してそれを用いて検索するのではなく、拡張語で STD を行った結果を用いて元のクエリの検索結果を調整することを指すこととする。

本論文では、STD におけるクエリ拡張について二種類の手法を提案する。1 つ目は、クエリの関連語を拡張語とする

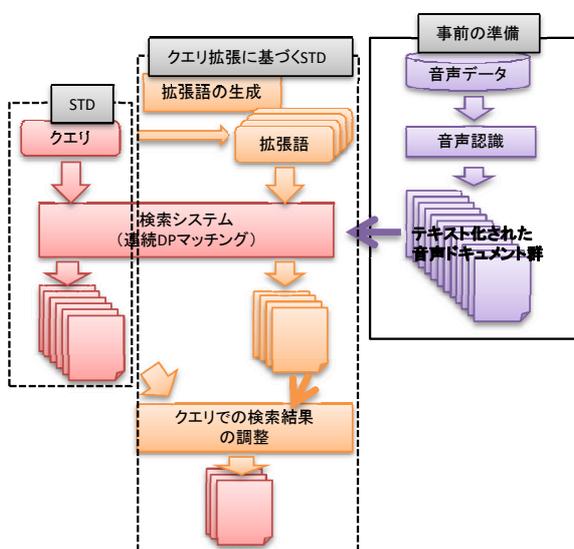


図 1 クエリ拡張に基づく STD の概要

る手法である。2 つ目は、クエリの前後に文字列を付加したものを拡張語とする手法である。本章では、これらについて詳細に述べる。

3.1 音声検索語検出とクエリ拡張

クエリ拡張に基づく STD の概観を図 1 に示す。アルゴリズムは以下のとおりである。

クエリ拡張に基づく STD のアルゴリズム

- (1) ユーザーからクエリ Q （長さ L ）を受け取る
- (2) クエリを拡張し、拡張語を複数得る $QE_i (i = 1 \dots N)$ 。
- (3) クエリで検索（STD）し、検索結果 (R_Q) を得る。その際、検索結果には音声ファイル名と何番目の発話かおよび、クエリとの編集距離の情報を付与する
- (4) 各拡張語で検索（STD）し、検索結果 (R_{QE}) を得る。ここでの検索結果は音声ファイル名のリストとする
- (5) R_Q のうち、音声ファイル名が R_{QE} に含まれない、すなわち元のクエリと拡張語が同じ音声ファイルに見つからなかった場合は、編集距離にペナルティ p を加える
- (6) 調整後の編集距離を用いて、元のクエリ（長さ L ）と各音声ファイルとの距離の近さを表すスコア $(1 - \frac{\text{Modified Edit Distance}}{L})$ を計算し、その順に結果を出力する

本手法は、クエリと拡張語が同じ音声ファイルに出現しなかった場合に、当該ファイルの検索結果のスコアを低くする手法である。例えばクエリを「オバマ」（アメリカ大統領）、拡張語を「大統領」や「アメリカ」とした場合に、オバマが出現するものの、「大統領」や「アメリカ」が含まれない音声ファイルは、それらが含まれる音声ファイルと比べてアメリカのオバマ大統領についてのものではない可能性が高いため、当該音声ファイルの検索スコアを下げるという手法である。

次に、クエリ拡張のための拡張語の抽出方法について述べる。

3.2 関連語に基づく拡張語

はじめに、クエリと関連が深くよく共起する語（関連語）を拡張語とする方法について述べる。これにより、クエリと同音異義語が含まれる音声ファイルや、その検索語は発話されていないが音声認識によって誤って登録されているような音声ファイルが検索結果中に現れることを防ぐ効果が期待できる。

関連語抽出には様々考えられるが、本研究では wikipedia

を利用する．具体的には，当該クエリを見出しとする wikipedia のページの本文中でリンクが付与されている単語（日付と年号を除く）を，出現順に 10 件選んで関連語とみなし，これを拡張語として用いる．

3.3 クエリ前後に文字列を付加する拡張語

次に，クエリの前後につきやすい語を付加してクエリを長くし，これを拡張語とする方法について述べる．これにより，短いクエリが別の語の一部にマッチして誤検出されることを防ぐ効果が期待できる．また，音声認識誤りにより別の語が検索語として現れていたとしても，前後の文字列まで含めると拡張語にマッチしない場合には，それを検索語でないとみなすことも可能になると期待される．

前後にどのような文字列が現れやすいかを求める方法には様々考えられるが，本研究では，STD のタスクを考えたときに，クエリは基本的に名詞，特に固有名詞となることが多いことに着目し，クエリの前または後に格助詞（10 種類）^{*1}を付加して関連語とする．この手法には，どのようなクエリが与えられても拡張語を得ることができるという利点がある．

4. 評価実験

4.1 評価尺度

情報検索システムの検索性能の評価尺度には，正解が検索結果としてどれほど出力されたかを表す再現率（recall）と，検索結果中に正解がどの程度含まれているかを表す精度（precision）がある．理想的には，再現率と精度を同時に 1 に近づけることが望ましい．実際には両者はトレードオフの関係にあり，一般的に，検索結果を上位に絞ると再現率は低いものの精度が高く，検索結果を多く出力すると再現率は高くなるものの精度が低くなる，といった傾向がある．このため，ある検索出力数のときの再現率と精度だけで検索性能を評価するのは不十分である．この問題に対して，検索結果出力数を変化させて様々な再現率レベルのときの精度を求め，それらの平均をとった値が評価尺度として広く用いられる．この尺度により平均的に性能が高い検索システムを評価できる．このような評価尺度として，平均精度（Average Precision: AP）がある．

あるクエリ q に対する平均精度 AP_q は，式（1）で与えられる．

$$AP_q = \frac{1}{\#cor(q)} \sum_{t=1}^{N_q} (IsTrue(q, t) \cdot P(q, t)) \quad (1)$$

ここで， $\#cor(q)$ はクエリ q に対する正解文書数， N_q は検索システムがクエリ q の答え（検索結果）として出力した文書数， $IsTrue(q, t)$ はクエリ q での検索結果の t 番目

が正解であれば 1，そうでなければ 0 を返す関数であり， $P(q, t)$ は q の検索結果の t 番目までを評価したときの精度（precision）である．

この AP_q を全クエリで平均したものの（式（2））は，MAP（Mean Average Precision）とよばれる．MAP は 0 から 1 をとり，1 に近いほど平均的に精度が高いことを表す．本研究では MAP を評価尺度として用いる．

$$MAP = \frac{1}{N} \sum_q AP_q \quad (2)$$

4.2 実験

実験データには，NTCIR-9 SpokenDoc[10] のテストコレクションを用いた．これは日本語話し言葉コーパス（CSJ: Corpus of Spontaneous Japanese）[11] の講演音声を対象とした音声ドキュメント検索のためのテストコレクションである．

NTCIR-9 SpokenDoc では，STD タスクとして，検索対象を全講演（2702 講演）とする ALL タスクと一部（177 講演）を対象とする CORE タスクが設定されており，本研究では ALL タスクを用いた．クエリには dry run 用クエリ（100 件）と formal run 用クエリ（50 件）があり，ここでは dry run 用クエリ（100 件）を用いた．検索対象の講演音声の認識結果には，タスクオーガナイザから配布されているマッチドモデルによる単語音声認識結果（Word Corr.=74.1%，Word Acc.=69.2%，Syll. Corr.=83.0%，Syll. Acc.=78.1%）[10] を用いた．

4.3 関連語によるクエリ拡張の効果

関連語によるクエリ拡張を行った．関連語とクエリが同じ音声ファイルに現れなかった場合のペナルティ（クエリ拡張に基づく STD のアルゴリズムの手順（5）の p ）を， $p = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5$ として実験を行った．検索結果としてスコアの高い順に上位 1000 件を出力し，MAP を求めた．拡張語による検索（STD）では，編集距離 0 のもののみを出力した．

結果を表 1 に示す．クエリでの連続 DP マッチング（従来手法）による検索性能（MAP）は，0.616 であった．関連語によるクエリ拡張では，編集距離ペナルティが 0.5，1.0，1.5 では MAP が向上したもののそれよりも大きなペナルティを与えたときは MAP が低下した．これは，クエリと関連語が共起しない音声ファイルにも，ある程度の正解が含まれているためと考えられる．この原因として，関連語が適切に抽出されていない可能性があげられる．

各クエリの結果を確認したところ，100 件のクエリのうち wikipedia に見出しが存在するクエリは 53 件で，47 件のクエリではクエリ拡張が行われていなかった．この 53 件での評価を表 2 に示す．MAP が 0.585 から 0.600 と向

*1 日本語百科大辞典 [縮刷版][9] 225 ページ記載の 10 種類（が・の・に・を・へ・と・で・より・から・や）

表 1 関連語によるクエリ拡張結果 (MAP)

従来手法	関連語クエリ拡張	
	penalty p	MAP
0.616	0.5	0.624
	1.0	0.620
	1.5	0.620
	2.0	0.601
	2.5	0.610
	3.0	0.595
	3.5	0.604

表 2 関連語によるクエリ拡張の効果 (MAP)

従来手法	53 件	100 件
従来手法	0.585	0.616
関連語クエリ拡張 (penalty $p=0.5$)	0.600	0.624

53 件: wikipedia に見出しが存在するクエリ

上していることがわかる。53 件のうち精度が向上したクエリは 39 件、低下したクエリは 2 件、精度の変化がなかったクエリは 12 件であった。

100 件のクエリのうち 39 件でしか向上がみられなかった (59 件では精度が変わらなかった) ことから、関連語の抽出方法に課題が大きいことがわかった。また、関連語については編集距離 0 のもののみしか検出していないため、この点も工夫の余地があると考えられる。

これらの結果は、関連語が正しく抽出できれば、関連語によるクエリ拡張が有効であることを示している。

4.4 クエリ前後に文字列を付加するクエリ拡張の効果

次に、クエリの前または後に文字列を付加するクエリ拡張の評価を行った。本研究では、前に格助詞をつけた 10 語を拡張語とする場合、後ろにつけた 10 語とする場合、これらの両者 20 語とする場合、の 3 通りを試した。

ここでも拡張語とクエリが同じ音声ファイルに現れなかった場合のペナルティ (クエリ拡張に基づく STD のアルゴリズムの手順 (5) の p) を、 $p = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5$ として実験を行った。検索結果としてスコアの高い順に上位 1000 件を出力し、MAP を求めた。拡張語による検索 (STD) では、元のクエリによる検索結果中の最小編集距離を求め、その編集距離のもののみを出力した。すなわち、元のクエリで連続 DP を行ったときの最小編集距離が l であったとき、拡張語による STD では編集距離 l のもののみを出力した。

結果を表 3 に示す。3 通り全てで MAP の向上がみられた。クエリの前または後に格助詞をつけた合計 20 語を拡張語として用いた場合 (ペナルティ=2.5) の時に最も高い精度 (MAP=0.700) が得られた。ペナルティを 2.0 や 3.0 にした場合は、他のペナルティのときに比べて相対的に精度が低かった。これは、クエリが編集距離 N で検出されかつ拡張語がみつからなかった場合と、クエリが編集距離

表 3 前後に文字列を付加するクエリ拡張結果 (MAP)

従来手法	文字列付加クエリ拡張			
	penalty	後のみ	前のみ	前と後の併用
0.616	0.5	0.661	0.655	0.666
	1.0	0.663	0.635	0.670
	1.5	0.684	0.669	0.694
	2.0	0.637	0.613	0.659
	2.5	0.686	0.664	0.700
	3.0	0.628	0.592	0.658
	3.5	0.675	0.647	0.694

表 4 文字列を付加するクエリ拡張の効果 (クエリ長での比較)
拡張: 前後の併用, penalty=2.5

クエリ長	8 音素以下	9~12 音素	13 音素以上
クエリ数	28	34	38
従来	0.517	0.608	0.696
拡張	0.613	0.674	0.786

$N + m$ ($m = 2.0, 3.0$) で検出されかつ拡張語が見つかった場合の検出結果を、同じ検索順位として扱うことが適切でないことを示している。

精度が最も高かった前後の併用 (ペナルティ 2.5) のときの検索精度とクエリ拡張なしのときの検索精度を、100 件のクエリそれぞれについて比較した。結果を図 2 に示す。また、クエリ長に基づいてクエリを 3 グループに分けて評価した結果を表 4 に示す。図 2 より、82 件のクエリで精度が向上し、13 件のクエリで精度が低下した (5 件は変化なし) ことがわかる。大きな精度向上がみられたクエリが多数確認できるのに対して、大きく精度が低下するクエリは存在しないことが確認できる。表 4 より、クエリ長によらず、提案法により MAP が 0.1 ポイント程度向上していることがわかる。これらのことは、前後に格助詞を付加するクエリ拡張の有効性を示している。

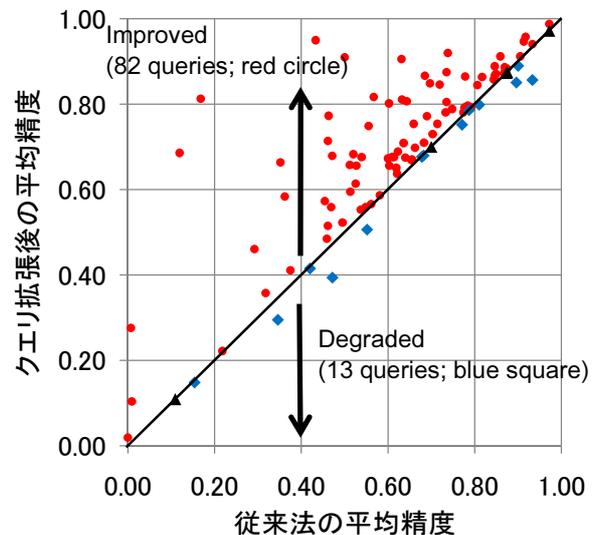


図 2 文字列を付加するクエリ拡張の効果 (クエリごとの分析)
拡張: 前後の併用, penalty=2.5

5. 他タスクでの評価実験

4章の結果より, STDにおいてクエリ拡張に効果があることがわかった. 特に, 前後に格助詞を付加するクエリ拡張が効果的であることを確認した. 本章では, 最も精度が高かったクエリ拡張手法, すなわち前後に格助詞を付与したものの併用(ペナルティ 2.5)を用いて, 種々のタスクでクエリ拡張の効果を調べる.

5.1 NTCIR-9 SpokenDoc formal run

NTCIR-9 SpokenDoc formal run で評価を行った. これは, NTCIR-9 SpokenDoc dry run と検索対象が同じ(CSJの全ての講演のマッチドモデルによる単語音声認識結果: Word Corr.=74.1%, Word Acc.=69.2%, Syll. Corr.=83.0%, Syll. Acc.=78.1%)であり, クエリが異なるタスクである.

結果を表5に示す. 結果の傾向は dry run の結果と同じであり, クエリを変えても提案するクエリ拡張手法は効果があることが確認できた.

5.2 NTCIR-10 SpokenDoc2 formal run

NTCIR-10 SpokenDoc2[12] formal run で評価を行った. CSJの2702講演を検索する Large-size タスクと音声ドキュメント処理ワークショップ(第1回~第6回)での講演音声(104件)を検索する Moderate-size task があるため, この両方で評価を行った.

5.2.1 Large-size task

これは, 4章の実験と検索対象が同じ(CSJの全ての講演のマッチドモデルによる単語音声認識結果: Word Corr.=74.1%, Word Acc.=69.2%, Syll. Corr.=83.0%, Syll. Acc.=78.1%)であり, クエリが異なるタスクである.

結果を表5に示す. クエリには, 単語音声認識時に単語辞書に含まれる既知語クエリと, 含まれない未知語クエリが存在するため, それぞれでの評価も行っている. 既知語, 未知語に関わらずクエリ拡張に効果があることがわかる. 特に未知語クエリに対して, 精度の大きな向上(0.403から0.538)がみられた.

提案クエリ拡張法は既知語と未知語の両方に効果があること, および, 特に未知語に効果があることがわかった.

5.2.2 Moderate-size task

これは, 4章の実験とは, 検索対象もクエリも異なるタスクである. 検索対象は音声ドキュメント処理ワークショップの講演音声(104件)である. 音声認識結果として, タスクオーガナイザから提供された下記の4種類のもの[12]を用いた.

- (1) マッチドモデルによる単語音声認識結果
Word Corr.=68.4%, Word Acc.=63.1%
Syll. Corr.=79.7%, Syll. Acc.=75.3%

表5 NTCIR-9 SpokenDoc formal run ALL task での評価

従来手法	文字列付加クエリ拡張			
	penalty	後のみ	前のみ	前と後の併用
0.485	0.5	0.529	0.520	0.537
	1.0	0.537	0.526	0.547
	1.5	0.570	<u>0.555</u>	0.583
	2.0	0.528	0.521	0.546
	2.5	<u>0.575</u>	0.551	<u>0.589</u>
	3.0	0.537	0.515	0.554
	3.5	0.571	0.543	0.586

表6 NTCIR-10 SpokenDoc2 formal run Large-size task での評価(クエリ拡張: 文字列を前後に付加, ペナルティ=2.5)

全クエリ		既知語クエリ		未知語クエリ	
従来法	拡張	従来法	拡張	従来法	拡張
0.479	0.573	0.568	0.615	0.403	0.538

- (2) アンマッチドモデルによる単語音声認識結果

Word Corr.=48.4%, Word Acc.=43.7%

Syll. Corr.=67.8%, Syll. Acc.=62.8%

- (3) マッチドモデルによる音節音声認識結果

Syll. Corr.=72.7%, Syll. Acc.=67.7%

- (4) アンマッチドモデルによる音節音声認識結果

Syll. Corr.=60.3%, Syll. Acc.=55.2%

なお, この検索対象の音声認識精度は4章での対象(CSJ)よりも低い.

結果を表7に示す. ここでも既知語と未知語のそれぞれの結果も示してある. なお, 音節認識では既知語と未知語の区別はないが, 単語認識での既知語/未知語の区分をそのまま使用している. これは, 既知語のSTDでは単語認識結果を使って, 未知語のSTDでは音節認識結果を使うといった使い方が想定されるためである.

全ての実験において, 提案法の有効性が確認できた. 既知語については, 単語音声認識結果を対象にSTDを行った場合に検索精度が高く, 提案法でさらに改善が得られていることがわかる. 未知語については, 音節音声認識結果を対象にSTDを行った場合に検索精度が高く, 提案法でさらに改善が得られていることがわかる.

これらのことより, 既知語のSTDでは単語認識結果を使い, 未知語のSTDでは音節認識結果を使うSTDでも本手法は有効であることがわかった.

5.3 評価実験のまとめ

種々のタスクで提案法の有効性が確認できた. 提案法について, 異なる検索対象, クエリでも効果的であること, 既知語, 未知語のどちらにも効果的であること, 音声ドキュメントの音声認識精度が異なる場合でも効果がみられること, を確認した.

表 7 NTCIR-10 SpokenDoc2 formal run Moderate-size task での評価 (クエリ拡張: 文字列を前後に付加, ペナルティ=2.5)

音声ドキュメント: 単語音声認識 (マッチドモデル)

全クエリ		既知語クエリ		未知語クエリ	
従来法	拡張	従来法	拡張	従来法	拡張
0.338	0.396	0.509	0.552	0.187	0.258

音声ドキュメント: 単語音声認識 (アンマッチドモデル)

全クエリ		既知語クエリ		未知語クエリ	
従来法	拡張	従来法	拡張	従来法	拡張
0.305	0.365	0.380	0.432	0.239	0.306

音声ドキュメント: 音節音声認識 (マッチドモデル)

全クエリ		既知語クエリ		未知語クエリ	
従来法	拡張	従来法	拡張	従来法	拡張
0.296	0.360	0.354	0.430	0.245	0.299

音節認識のため既知語と未知語の区別はない。
 単語認識での既知語/未知語の区分

音声ドキュメント: 音節音声認識 (アンマッチドモデル)

全クエリ		既知語クエリ		未知語クエリ	
従来法	拡張	従来法	拡張	従来法	拡張
0.313	0.366	0.343	0.398	0.287	0.338

音節認識のため既知語と未知語の区別はない。
 単語認識での既知語/未知語の区分

6. おわりに

STDにおけるクエリ拡張の研究を行った。ユーザが入力した初期クエリに対していくつかの拡張語を生成し、その拡張語でSTDを行った結果を用いて元のクエリの検索結果を調整する方法を研究した。拡張語の獲得方法として、クエリの関連語を拡張語とする手法とクエリの前後に文字列を付加したものを拡張語とする手法を提案した。

講演音声を対象としたSTDにより、関連語を拡張語とする手法およびクエリの前後に文字列(格助詞)を付加して拡張語とする方法ともに効果があることを示した。後者の提案法については、複数のSTD検索タスクで検索精度の大きな向上が得られることを示した。提案法が、異なる検索対象、クエリでも効果をもつこと、既知語、未知語のどちらにも効果をもつこと、音声ドキュメントの音声認識精度が異なる場合でも効果をもつことを明らかにした。

謝辞 本研究は科研費の助成を受けた。

参考文献

[1] Akiba, T., Aikawa, K., Itoh, Y., Kawahara, T., Nanjo, H., Nishizaki, H., Yasuda, N., Yamashita, Y. and Itou, K.: Construction of a Test Collection for Spoken Document Retrieval from Lecture Audio Data, *IPSJ-journal*, Vol. 50, No. 2, pp. 82-94 (2009).

[2] 南條浩輝, 弥永裕介, 吉見毅彦: 広域文書類似度と局所

文書類似度を用いた講演音声ドキュメント検索, 情報処理学会論文誌, Vol. 53, No. 6, pp. 1654-1662 (2012).

[3] 西尾友宏, 南條浩輝, 吉見毅彦: 講演音声ドキュメント検索のための擬似適合性フィードバック, 情報処理学会論文誌 (採録決定), Vol. 55, No. 5 (2014).

[4] 伊藤慶明, 西崎博光, 中川聖一, 秋葉友良, 河原達也, 胡新輝, 南條浩輝, 松井知子, 山下洋一, 相川清明: 音声中の検索語検出のためのテストコレクションの構築と分析, 情報処理学会論文誌, Vol. 54, No. 2, pp. 471-483 (2013).

[5] Noritake, K., Nanjo, H. and Yoshimi, T.: Image Processing Filters for Line Detection-based Spoken Term Detection, *Proc. INTERSPEECH*, pp. 2125-2128 (2011).

[6] Natori, S., Furuya, Y., Nishizaki, H. and Sekiguchi, Y.: Spoken Term Detection Using Phoneme Transition Network from Multiple Speech Recognizers' Outputs, *Journal of Information Processing*, Vol. 21, No. 2, pp. 176-185 (2013).

[7] 大野哲平, 秋葉友良: 音節継続時間を利用した直線検出に基づく音声検索語検出, 情報処理学会論文誌, Vol. 54, No. 2, pp. 484-494 (2013).

[8] Nanjo, H., Noritake, K. and Yoshimi, T.: Spoken Document Retrieval Experiments for SpokenDocat Ryukoku University (RYS DT), *NTCIR-9 Workshop Meeting*, pp. 275-280 (2011).

[9] 金田一春彦, 林大, 柴田武: 日本語百科大辞典 [縮刷版], 大修館書店. ISBN: 4-469-01244-0.

[10] Akiba, T., Nishizaki, H., Aikawa, K., Kawahara, T. and Matsui, T.: Overview of the IR for Spoken Documents Task, *NTCIR-9 Workshop Meeting*, pp. 223-235 (2011).

[11] Maekawa, K.: Corpus of Spontaneous Japanese: Its design and evaluation, *Proc. ISCA & IEEE-SSPR*, pp. 7-12 (2003).

[12] Akiba, T., Nishizaki, H., Aikawa, K., Hu, X., Itoh, Y., Kawahara, T., Nakagawa, S., Nanjo, H. and Yamashita, Y.: Overview of the NTCIR-10 SpokenDoc-2 Task, *NTCIR-10 Workshop Meeting*, pp. 573-587 (2013).