Vol.2012-MUS-94 No.23
Vol.2012-SLP-90 No.23
2012/2/4

音声書き起こし支援システムに向けた自動頭出し機能の開発と評価

芦川 平[†1] 永尾 学[†1]
西山 修[†1] 池田朋男[†1]

録音された音声を聞きながら、発話の内容を文字に書き起こす作業（書き起こし作業）において、音声認識技術を利用し、発話内容を自動で文字に書き起こす「自動書き起こし」の実現が望まれている．しかし、現在の音声認識技術において、発話者や環境が限定されていない場合に、精度の高い音声認識結果を得ることは難しい．一方、現在の書き起こし作業においては、書いている位置と再生位置のずれが発生するため、再生位置を手動で戻すという行為が多く発生し、作業者の負担が大きい．そこで、本報告では、書き起こし作業の支援として、作業者の書き起こし状況に応じた音声の再生制御を行う機能を検討する．ラティスおよび強制アライメントを用いた頭出し位置の推定アルゴリズムを検討し、実際の会議等での音声データを用いた評価実験を行い、提案手法が有効であることを確認する．

An estimation algorithm of the play position for the support system of dictating

Taira Ashikawa,[†1] Manabu Nagao,[†1]
Osamu Nishiyama[†1] and Tomoo Ikeda[†1]

Auto dictation systems are required using the speech recognition technology for dictation workers. However it is difficult to get correct answers under the various circumstances by the current SR technology. And now, the workers have the burden of moving the play position many times while dictating, because the position is farther than the next writing position due to the delay. We offer the automatic play function which controls the position depending on the user's current writing text in order to support the work. This report proposes the algorithm for searching the position using both lattice and forced alignment method and shows the evaluation result of the algorithm.

1. はじめに

書き起こし作業とは、録音された音声を聞きながら、発話の内容を文字に書き起こす作業のことである．この書き起こし作業において、音声認識技術を利用し、発話内容を自動で文字に書き起こす「自動書き起こし」の実現が望まれている．現在の音声認識技術において、発話者や環境が限定されている場合には、精度の高い認識結果を得ることは可能である．しかし、発話者や環境が限定されていない場合には、精度の高い音声認識結果を得ることは難しい．また、一般的に書き起こしをする録音ファイルは多種多様であり、発話者や環境は限定できない．そのため、汎用化された自動書き起こしの実現は難しく、現状では人手で書き起こしを行うことが多い．

また、音声コーパスの収集を効率化するために、音声データから文字を書き起こす入力ツールが研究されている[1)2)]．また、近年では、Crowd-Sourcing を用いて、より高品質なコーパスを大量に収集する方法も多く研究されている[3)]．しかし、上記の文献で報告されている Transcriber[1)]、BritzScriber[2)] は、音声コーパスを収集することが目的であるため、一般的な書き起こし作業には向いておらず、書き起こしを支援する機能は提供されていない．

一方、現在の書き起こし作業において、音声再生中に書き起こしが間に合わず、書き起こしている部分よりも音声が先に進んでしまう場合が多い．その場合、作業者自身が書き起こしていない位置まで音声データを手動で戻して、書き起こしを再開するという動作が発生し、作業者の負担が大きい．

そこで、我々は、音声認識技術を活用した書き起こし支援システムを開発した．本システムでは、書き起こし支援機能として、作業者の書き起こし状況に応じた音声の再生制御を行う機能 (自動頭出し機能)、話者を識別する機能 (話者識別機能)、書き起こした文中に存在する誤りを指摘し、ユーザの校正作業を支援する機能 (文章整文支援機能) などの機能を提供する．本報告では、主に自動頭出し機能について説明する．

まず、2 章で書き起こし支援システムと自動頭出し機能の概要を述べ、3 章で頭出し位置の推定アルゴリズムの検討、4 章で評価実験について述べ、最後に今後の課題を述べる．

†1 東芝 研究開発センター
Research & Development Center, Toshiba Corporation

Vol.2012-MUS-94 No.23
Vol.2012-SLP-90 No.23
2012/2/4

2. 書き起こし支援システム

我々が開発した書き起こし支援システムは、音声処理を用いることにより、ユーザの書き起こしの作業負担を軽減する支援機能を提供する．また、ユーザが特別なアプリケーションをインストールしなくても利用可能にすることと、Crowd-Sourcing による書き起こしを可能にするため、ウェブアプリケーションとして開発した．

本システムでは、ユーザが登録した音声ファイルに対して、まずノイズ除去、話者識別、ラティス生成等の前処理を行う．次に、ユーザはウェブブラウザを用いて、図 1 の書き起こし画面 (エディタ画面) にアクセスし、書き起こしを開始する．

エディタ画面では、上部に音声を制御する「音声制御部」と、下部に書き起こしを行う「エディタ部」が表示される．「音声制御部」では、音量設定、再生速度、タイムバー等を表示し、「エディタ部」では、発話時間ごとに区間 (話者区間) を設け、話者区間毎に発話開始時刻、話者名、書き起こし文を表示する．音声制御部のタイムバー上には、現在再生している音声位置 (音声再生位置) と、エディタ部の書き起こし文内のカーソル位置に対応する位置 (頭出し推定位置) を表示する．音声制御部の頭出し推定位置は、エディタ部のカーソル位置と同期して表示し、ユーザの特定のキー操作により、その位置から音声を再生する．ユーザの書き起こし時の操作の流れを下記に示す．

(1) ユーザが、音声制御部で音声を再生する
(2) ユーザは、音声を聞きながら、エディタ部で音声の内容を書き起こす
(3) システムは、ユーザが書き起こした文とカーソル位置から、頭出し推定位置を求める
(4) ユーザは、書き起こしをしている際に、音声を戻したい場合、Return ボタンを押下する
(5) システムは、頭出し推定位置から、音声を再生する
(6) ユーザは、同じ話者の書き起こしが終わるまで、(2)-(5) を繰り返す
(7) ユーザは、話者が切り替わった場合に、Ctrl-Return を押下する
(8) システムは、推定した話者開始時刻と話者名を持つ新しい話者区間を表示する
(9) ユーザは、すべての書き起こしが終了するまで、(2)-(8) を繰り返す

上記の機能を開発することにより、ユーザは書き起こし中にマウスを利用せず、キータイピングのみに集中して書き起こしができるため、書き起こしにかかる時間を短縮することができると考えられる．

図 1 エディタ画面
Fig. 1 system editor image

2.1 自動頭出し機能の基本要求

上記の流れを実現するため、本システムでは、以下の要求を満たす自動頭出し機能を開発する．

(1) 書き起こしが終わった近辺から音声を再生する
前述の通り、書き起こし作業において、音声再生中に書き起こしが間に合わず、書き起こしている部分よりも音声が先に進んでしまう場合が多い．そこで、本機能では、作業者が書き起こした文字に該当する音声位置をシステムが自動で推定し、作業者がある指定の動作を行う (Enter キーを押下する等) と、その推定位置から再生する．これにより、書き起こし中に音声が進みすぎた場合も、簡単な動作で書き起こしていない箇所から音声を再生することができる．

(2) 指定のカーソル位置に対応する箇所から音声を再生する
書き起こし作業において、聞き取りにくい箇所は、一通り書き起こしを行った後、聞き取りにくかった箇所の音声を再度聞き、書き起こすことも多い．その場合、聞き取りにくかった箇所まで音声を戻して再度聞く必要がある．そこで、作業者がすでに書き起こした文字に対して、カーソル位置を指定して、ある指定の動作を行うと、カーソル位置に該当する音声位置をシステムが自動で推定し、その推定位置から再生する．これにより、再度聞きたい個所にカーソルで位置を指定して、その場所から再生

Vol.2012-MUS-94 No.23
Vol.2012-SLP-90 No.23
2012/2/4

することができる．

(3) 不明箇所を表す記号の挿入や、倒置が行われた場合にも対応できる
実際の書き起こし作業では、一度聞いて書き起こせない場合は、不明な箇所として特定の文字 (例えば●など) を挿入しておき、後で再度聞いて書き起こすことがある．また、音声の通り書き起こすのではなく、読み手が読んで理解し易いように、話している単語や内容の順番を倒置して書き起こすこともある（例えば、「晴れていますね、今日は」を「今日は、晴れていますね」など）．これらの場合にも対応することが必要である．

2.2 関連研究

音声データから文字の出現位置を推定するアルゴリズムの関連研究として、音声ドキュメント検索という研究がある[4]．音声ドキュメント検索とは、音声データと検索クエリが入力として与えられ、検索クエリに適合する音声データの位置を特定することであるが、現在の典型的な解法は、1. 音声データの音声認識、2. 認識結果に索引付け、3. テキスト検索の処理を行い、音声位置を検索する．しかし、前述の通り、本システムにおける録音環境は多種多様であるため、1. 音声認識処理が常に精度が高い認識結果が得られるとは限らない．また、2. の索引付けに関しても、本システムの自動頭出し機能の要求を満たすだけに、索引付けを行うのはコストが高く、また、索引付けが終了するまで書き起こしができなくなるため、作業者にとって不便であると考えられる．

3. 頭出し位置推定アルゴリズムの検討

本章では、自動頭出し機能の頭出し位置推定のための独自のアルゴリズムを検討する．まず、ラティスを用いた推定方法と、強制アライメントを用いた推定方法について、アルゴリズムとその利点・欠点を述べる．その後、最終的に採用した両方式を統合した推定方法について説明する．

3.1 ラティス方式

前述の通り、現在の音声認識技術において、発話者や環境が限定されていない場合に、精度の高い音声認識結果を得ることは難しい．そこで、頭出し位置の推定には、最終的な音声認識結果ではなく、認識途中で出力されるラティスを用いて位置を推定する方法を提案する．ラティス (単語ラティス) とは、音声認識結果の候補単語を連結したグラフのことであり、音声認識においては、一般的に最終出力の前の候補として生成される[5][6]．

ラティスのノード数とアーク数は、音声認識エンジンのパラメタにより増減することは可能である．ラティス上に必ずしも正解がある (書き起こし文字と候補文字が一致する) とは限らないが、候補数が多い方が正解する確率は高いと考えられる．また、自動頭出し機能の頭出し位置の推定は、リアルタイムで実行できなければならない．しかし、ラティスに存在するアーク上の候補語のすべての組み合わせを探索した場合、組み合わせ爆発が発生し、リアルタイムでの探索は困難である．そこで、ラティスを用いた頭出し位置推定のアルゴリズムを、次のようにする．

[アルゴリズム]

(1) エディタ上のカーソルがある話者区間の書き起こし文字列 $InputString$、現在のカーソル位置 $CursorPosition$、音声開始位置 $SoundStartPosition$、現在の音声再生位置 $SoundCurrentPosition$ を取得する

(2) $SoundStartPosition$ から $SoundCurrentPosition$ を推定区間 S とする

(3) 推定区間 S 内に存在するラティスのアーク群 $Archs$ を取得する

(4) $InputString$ 内の $CursorPosition$ 番目の語が含まれる文 $Sentence$ を抽出する

(5) $Archs$ の中で、$Sentence$ に存在する語を候補語として持つアーク $PredictArchs$ を抽出する．この際、候補語が 2 文字以上のアークのみを抽出する

(6) $PredictArchs$ の中で、右ノードの出現時間が最大であるアーク $MaxPredictArch$ を抽出する

(7) $Archs$ の中で、$MaxPredictArch$ の候補語と同じ候補語を持つアークが存在しなければ、$MaxPredictArch$ を候補アーク $PredictArch$ とする．$MaxPredictArch$ の候補語と同じ候補語を持つアークが複数存在する場合は、右ノードの出現時間が最小のアークを候補アーク $PredictArch$ とする．

(8) $PredictArch$ に隣接する右ノードの出現時間を、頭出し推定位置 $PredictPosition$ とする

[利点]

- 不明語が挿入されている場合にも対応できる
- 倒置が行われている場合にも対応できる
- 探索における組み合わせ爆発は起こらない

前述の通り、実際の書き起こし作業において、一度聞いて書き起こせない場合は、不明な箇所として特定の文字 (例えば●など) を挿入 (不明挿入) や、単語の順番を倒置して書き起こすことがある．上記アルゴリズムの場合、書き起こし文字列に存在する語を持つアークを探索し、それらのアークの中で最大出現時間を持つアークを推定位置とするため、不明挿入や

倒置の場合にも対応できる．また、アークの深さ探索や幅探索しないため、組み合わせ爆発は起こらず、探索時間は推定区間内のアーク数のみに比例する．

[**欠点**]

- 書き起こし文字列とラティスの候補文字列が一つも一致しない場合には、頭出し位置を推定できない

上記アルゴリズムは、ラティスのみを利用しているため、ラティスの候補文字に書き起こし文字が存在しない場合には、頭出し位置を推定できない．

3.2 強制アライメント方式

上記のラティス方式での欠点に対処するため、別の音声認識技術である強制アライメント (フォースドアライメント) を利用する．強制アライメントとは、発話単位で区切られた音声データと、その音声データに対応する単語列を用いることにより、音声データを構成する各単語の時刻情報を得るものである[7][8]．強制アライメントを用いた頭出し位置推定のアルゴリズムを、次のようにする．

[**アルゴリズム**]

(1) エディタ上のカーソルがある話者区間の書き起こし文字列 $InputString$、現在のカーソル位置 $CursorPosition$、発話開始位置 $SpeakStartPosition$、現在の音声再生位置 $SoundCurrentPosition$ を取得する
(2) $SpeakStartPosition$ から $SoundCurrentPosition$ を推定区間 S とする
(3) $InputString$ の $CursorPosition$ 番目までの文字列を単語で区切り、単語ごとの読みを与えた文字列 $ReadingString$ を生成する
(4) 推定区間 S の間の音声データと、$ReadingString$ に対して、強制アライメントを行い、各単語の出現位置 $WordTimeInfo$ を取得する
(5) $WordTimeInfo$ から、最後の単語の出現時間を取得し、頭出し推定位置 $PredictPosition$ とする

[**利点**]

- 単語の出現位置を強制的に割り当てるので、ラティスの候補文字として出現しない書き起こし文字でも、単語の出現時間が取得できる

[**欠点**]

- 不明挿入、倒置に対応できない
- 処理に時間がかかる

前述のとおり、書き起こし作業の場合、不明文字の挿入や、単語の倒置が発生する．また、強制アライメントとは、そもそも、出現した単語の出現位置を求めるものであり、単語の出現順序が一致しない場合には利用できない．そのため、話した順番通りに書き起こしが行われない場合には、強制アライメントは、頭出し位置の推定には利用できない．また、強制アライメントの場合、ラティス方式と違い事前に情報を生成することができないため、入力がある度に、都度処理を実行しなければならない．そのためラティスに比べ、処理に時間がかかってしまう．

3.3 ハイブリッド方式

上記、ラティス方式、強制アライメント方式の利点と欠点を踏まえ、ラティス方式と強制アライメント方式の両方を合わせた推定方法を、ハイブリッド方式として採用する．ハイブリッド方式の頭出し位置推定のアルゴリズムを、次に示す．

[**アルゴリズム**]

(1) エディタ上のカーソルがある話者区間の書き起こし文字列 $InputString$、現在のカーソル位置 $CursorPosition$、発話開始位置 $SpeakStartPosition$、音声開始位置 $SoundStartPosition$、現在の音声再生位置 $SoundCurrentPosition$ を取得する
(2) ラティス方式で頭出し位置を推定する (ラティス方式の (2)〜(8) を実行)
(3) (2) で推定位置を取得できない場合、強制アライメント方式で頭出し位置を推定する (強制アライメント方式の (2)〜(5) を実行)
(4) (3) で推定位置が取得できない場合は、$SoundCurrentPosition$ から一定時間戻した位置を推定位置 $PredictPosition$ とする

4. 評価実験

本章では、3 章で検討した頭出し位置推定アルゴリズムについての評価実験を行い、その効果を示す．

4.1 実験条件

評価実験には、表 1 の 6 つの音声データを利用した．いずれのデータも 10 分程度であり、市販の IC レコーダで録音した音声データである．

2.1 節で記述したとおり、書き起こしにおける頭出しでは、作業者が書き終わった文字の後ろの文字が出現する位置から音声が再生されることが望ましい．そこで、評価指標として、入力された文字列から推定された頭出し位置と、入力された文字の次の語の出現位置との時間差 d_p を以下の式で求め、評価を行うこととした．ただし、$s(i)$ =i 番目の語の出現位置, $p(i)$ =i 番目の語の後ろにカーソル位置があった場合の頭出し推定位置とする．

Vol.2012-MUS-94 No.23
Vol.2012-SLP-90 No.23
2012/2/4

表 1 実験データ
Table 1 experimet data

データ ID	環境	話者数	出現単語数
data1	会議室	4	701
data2	会議室	3	1257
data3	会議室	6	836
data4	展示会会場	2	1230
data5	ラジオ	2	939
data6	ラジオ	10	1008

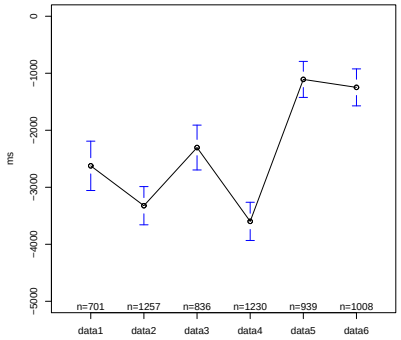

図 2 語の出現位置と推定位置の時間差 d_p
Fig. 2 time difference between appearance position and estimated position

$$d_p(i) = s(i) - p(i-1) \quad (1)$$

そこで、語の出現位置 $s(i)$ を収集するため、まず、data1〜data6 の音声データについて、人手で書き起こしを行う．次に、書き起こした文章に対して、形態素解析を行い語を抽出し、最後に人が音声データを耳で聞き、二文字以上の各語の出現位置を記録し、$s(i)$ を収集した．各データの二文字以上の出現単語数は、表 1 の通りである．

また、3 章で説明したように、頭出し位置推定アルゴリズムには、音声開始位置と現在の音声再生位置を利用する．今回の評価実験では、語の出現位置から 5 秒引いた位置を音声開始位置とし、語の出現位置から 0〜20 秒の範囲でランダムに選択した位置を現在の音声再生位置とした．

4.2 実験結果

各音声データに対する、頭出し位置推定アルゴリズムを利用した場合の時間差 d_p の平均値と 95%信頼区間の下限値と上限値を図 2 に示す．図の通り、頭出し位置推定アルゴリズムを利用した場合、ユーザが書き起こしたい語の前で頭出しの操作を行った場合、data1〜data4 のデータに関しては、平均で約 3 秒後、data5,data6 に関しては、平均で約 1 秒後から音声が再生されることがわかった．ユーザにとっては、書く位置よりも前の位置から音声が再生された方が分かりやすいため、実際の頭出し機能では、推定値から 5 秒程度引いた値を利用するのが適切だと考えられる．

表 2 従来手法との差 d_d
Table 2 time difference between the simple mehtod and the estimate method

データ ID	平均値 (ms)	下限値 (ms)	上限値 (ms)
data1	2225.815	1894.939	2556.690
data2	2112.402	1865.094	2359.710
data3	2606.519	2276.548	2936.490
data4	1986.229	1741.487	2230.972
data5	3726.348	3406.863	4045.833
data6	3357.995	3052.701	3663.289

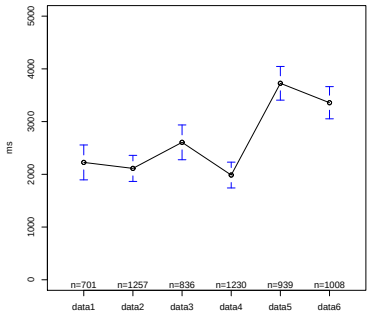

図 3 従来手法との差 d_d
Fig. 3 time difference between the simple mehtod and the estimate method

次に、従来の書き起こしツールでよく利用されている「特定のキーアクションにより再生位置から一定時間戻って再生する」手法と比較する．これは推定範囲の終端位置から定数時間戻った位置を推定位置とすることと同じであるため、以下の式 (2) で語の出現位置との時間差 d_c を求める．また、従来手法の時間差 d_c と、推定アルゴリズムの時間差 d_p との差を以下の (3) 式で求める．ここで、$se(i)$ =i 番目の推定範囲の終端位置、k=3000 とする．

$$d_c(i) = s(i) - (se(i-1) - k) \quad (2)$$

$$d_d(i) = |d_c(i)| - |d_p(i)| \quad (3)$$

表 2 と図 2 に、$d_d(i)$ の平均値と 95%信頼区間の下限値と上限値を示す．この結果から、頭出し位置推定アルゴリズムを利用した場合、従来の一定時間戻る手法と比べ、data1〜data4

Vol.2012-MUS-94 No.23
Vol.2012-SLP-90 No.23
2012/2/4

表 3 ハイブリッド方式内の認識割合
Table 3 percentage in hybrid method

データ ID	ラティス	強制アライメント	定数時間
data1	72.7%(-1843.47)	6.3%(-6813.26)	21.0%(-4784.91)
data2	72.7%(-2696.95)	9.4%(-6461.79)	17.9%(-4622.39)
data3	80.4%(-1674.36)	6.8%(-7085.39)	12.8%(-4673.54)
data4	55.7%(-2445.08)	12.8%(-5812.05)	31.6%(-5039.94)
data5	83.8%(-682.72)	9.9%(-1496.46)	6.3%(-7060.03)
data6	91.5%(-1058.73)	4.2%(61.92)	4.4%(-5557.71)

のデータに関しては、平均で約 2 秒、data5,data6 に関しては、平均で約 3 秒、実際の語の出現位置に近かった．t 検定を実施したところ、全ての音声データにおいて $p < 0.05$ となり、従来手法と比べて有意差が見られた．

次に、ハイブリッド方式の中で、ラティス方式、強制アライメント方式、一定時間戻す方式のどの方式で推定されたかの内訳を表 3 に示す．各方式の時間差 d_p の平均値をカッコ内に示す．また、図 4 にこれらの結果をまとめたグラフを示す．図 4 では、縦棒に各方式の割合、折れ線で時間差 d_p の平均値を示している．表 3 が示すように、ハイブリッド方式では、ラティス方式で平均で 76.1% の語が推定できていたが、ラティス方式で取れない残りの語に関しては、強制アライメント方式で 8.23% が推定できていた．強制アライメント方式と定数時間方式を比較した場合に、data1〜data4 に関しては、定数時間方式の方が出現位置との差が小さく良好であり、data5,data6 に関しては、強制アライメントの方が出現位置との差が小さく良好であった．

5. おわりに

本報告では、書き起こし作業支援として、作業者の書き起こし状況に応じた音声の再生制御を行う機能を提供するため、ラティスおよび強制アライメントの両方式を用いた頭出し位置の推定アルゴリズムを検討し、実際の会議を録音した音声データを用いて評価実験を行った．実験結果から従来手法よりも、頭出し位置を 2〜3 秒ほど近く推定することが確認できた．

今後の課題としては、強制アライメントを利用した場合に、音声データによっては良い結果を得られない、または、入力内容によっては処理時間がかかるため、実際に機能として提供する場合には十分な精度が得られない可能性がある．そこで、一度処理した結果はキャッシュしておくなどの、処理時間短縮の方法の検討が必要である．また、ユーザが自動頭出し

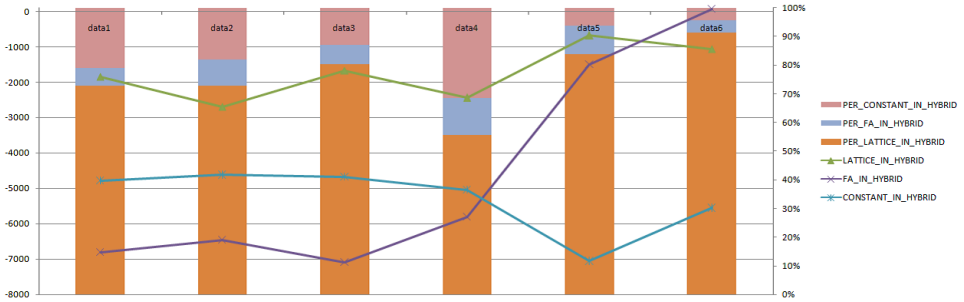

図 4 ハイブリッド方式内の認識割合
Fig. 4 percentage in hybrid method

機能を利用した場合に、書き起こしにかかる時間と正解率から本機能の検証を行いたい．

参 考 文 献

1) Barras, C., Geoffrois, E., Wu, Z. and Liberman, M.: Transcriber: Development and use of a tool for assisting speech corpora production, *Speech Communication*, Vol.33, No.1-2, pp.5 – 22 (2001).
2) Roy, O.C. and Roy, D.: Fast Transcription of Unstructured Audio Recordings, *Interspeech 2009*, pp.1647–1650 (2009).
3) Marge, M., Banerjee, S. and Rudnicky, A.: Using the Amazon Mechanical Turk for transcription of spoken language, *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp.5270 –5273 (2010).
4) 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. SLP, 音声言語情報処理， Vol.2010, No.10, pp.1–8 (2010).
5) 清水 徹，山本博史，松永昭一，匂坂芳典：単語グラフを用いた自由発話音声認識，情報処理学会研究報告. SLP, 音声言語情報処理， Vol.95, No.120, pp.49–54 (1995).
6) Ortmanns, S., Ney, H. and Aubert, X.: A word graph algorithm for large vocabulary continuous speech recognition, *Computer Speech Language*, Vol.11, No.1, pp. 43 – 72 (1997).
7) Moreno, P.J., Joerg, C., Thong, J.-M.V. and Glickman, O.: A recursive algorithm for the forced alignment of very long audio segments, *ICSLP-1998* (1998).
8) Haubold, A. and Kender, J.: Alignment of Speech to Highly Imperfect Text Transcriptions, *Multimedia and Expo, 2007 IEEE International Conference on*, pp.224 –227 (2007).