同時的な独話音声要約に基づくリアルタイム字幕生成

大野 誠寛† 松原 茂樹‡ 柏岡 秀紀§ 稲垣 康善♮

†名古屋大学大学院情報科学研究科 ‡名古屋大学情報連携基盤センター
§ATR 音声言語コミュニケーション研究所 ♮愛知県立大学情報科学部
E-mail: ohno@el.itc.nagoya-u.ac.jp

概要

講演や解説などの独話をリアルタイムに理解することを支援する字幕生成システムが切望されている．このリアルタイム字幕生成システムには，話者の発話内容を字幕の表示時間内に視聴者が理解できる程度に要約し，生成した字幕をその入力音声に追従して出力することが求められる．そこで，本稿では，節境界に基づく漸進的係り受け解析を利用した独話音声のリアルタイム字幕生成手法を提案する．本手法では，節が検出されるたびに同定される係り受け構造に基づいて，要約処理を実行することにより，入力音声のリアルタイムな字幕化を実現している．実際の独話データを用いた実験により，本手法の実現可能性を確認した．

キーワード 字幕生成, 漸進的解析, 係り受け解析, 独話, 音声言語

Real-time Captioning based on Simultaneous Summarization of Spoken Monologue

Tomohiro Ohno† Shigeki Matsubara‡ Hideki Kashioka§ Yasuyoshi Inagaki♮

†Graduate School of Information Science, Nagoya University
‡Information Technology Center, Nagoya University
§ATR Spoken Language Translation Research Laboratories
♮Faculty of Information Science and Technology, Aichi Prefectural University
E-mail: ohno@el.itc.nagoya-u.ac.jp

Abstract

The development of the captioning system, which supports the real-time understanding of monologue speech such as the lecture and commentary, is demanded. In the real-time captioning system, it is necessary to summarize the speech so that the audience can understand in the display time and to output the caption simultaneously with the monologue speech input. This paper proposes a technique for real-time captioning of spoken Japanese monologue using incremental dependency parsing based on clause boundaries. The technique identifies the summary unit and summarizes it based on the dependency structure identified whenever a clause boundary is detected. An experiment using Japanese monologues has shown the feasibility of our technique.

key words captioning, incremental parsing, dependency parsing, monologue, spoken language

1 はじめに

講演や解説などの独話をリアルタイムに理解することを支援する字幕生成システムの開発が望まれている．リアルタイム字幕生成システムでは，音声に追従して字幕を出力することがもとめられる．そのため，字幕の表示時間をいたずらに延長できないことから，聴衆が字幕テキストを読む速度を考慮した場合，冗長箇所を削除し要約する必要がある．一方，従来の音声要約手法では，文に対して要約処理が行われているが（例えば，[1]），独話には明示的な文末標識がなく，事前に文単位に区切ることは容易ではない．また，たとえ文境界を検出できたとしても，独話文は長くなる傾向にあるため，音声に追従して字幕を出力するという同時性を損なうことになる．このように，リアルタイムに音声を要約する場合，どのような言語単位を要約処理の単位として定めるかが問題となる．

そこで本稿では，節境界に基づく漸進的係り受け解析 [5] を利用した独話音声のリアルタイム字幕生

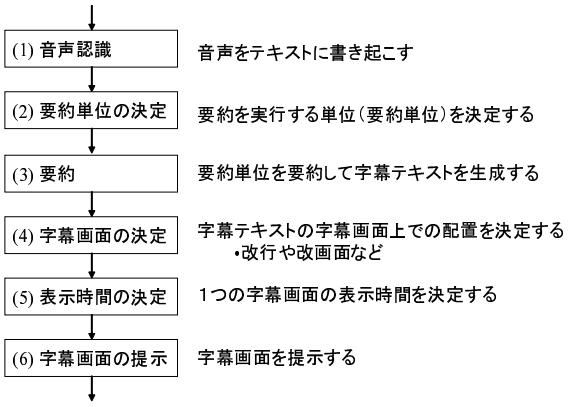

図 1: リアルタイム字幕生成システムの流れ

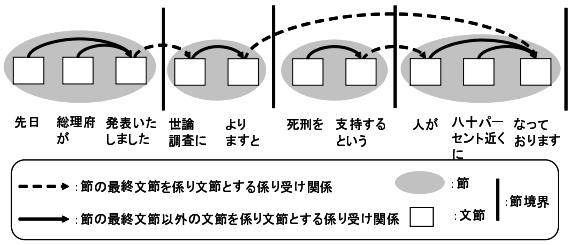

図 2: 節境界と係り受けの関係

成手法を提案する．本手法では，節が検出されるたびに同定される係り受け構造に基づいて，順次，要約処理を実行する単位を定め，その単位ごとに要約処理を実行する．これにより，入力音声のリアルタイムな字幕化の実現が期待できる．さらに，この単位は，同定された節末の係り受け関係によって連結された節列であるため，統語的，意味的にまとまっており，要約処理に適していると考えられる．また，要約では，係り受け構造を考慮して，冗長と思われる文節単位，節境界単位を順に削除することにより，不自然な日本語の生成を防いでいる．

本手法の有効性を評価するために，NHK の解説番組「あすを読む」の独話データを用いて字幕生成実験を行った．実験の結果，本手法の実現可能性を確認した．

2　リアルタイム字幕生成

一般に，リアルタイムに音声を要約し字幕を生成するためには，図 1 に示す処理を順に行う必要がある．

このうち，本研究では，(2)〜(5) を扱う．(4),(5) の字幕画面とその表示時間の決定は，字幕表示方法によって大きく影響される．まず，本研究では，字幕表示方法としてテレビのクローズドキャプションに見られる切り替え方式を採用する．ただし，一画面あたりの文字数の制限については考慮せず，要約結果のテキストをすべて字幕の一画面として表示する．

また，字幕の表示時間は，その元となる要約単位の発話時間分だけ表示する．なお，表示開始は，字幕画面が生成された時点で，一つ前の字幕画面の表示が終了していればすぐに行い，終了していなければ一つ前の字幕画面の表示が終了次第行う．

3　要約単位の決定

本研究では，要約処理を実行する単位を節境界に基づく漸進的係り受け解析 [5] の結果に基づいて決定する．以下では，まず，節境界に基づく漸進的係り受け解析手法について概説し，次に，要約単位決定アルゴリズムについて述べ，最後に，解析例を示す．

3.1　節境界に基づく漸進的係り受け解析

本手法では，解析の処理単位として節を採用し，節が入力されるたびにその時点までの入力に対して係り受け解析を実行する．節とは，述語を中心としたまとまりであり，複文や重文の場合，文は複数の節から構成される．さらに，節は，統語的，意味的にまとまった単位であるため，文に代わる解析単位として利用できる．例として，独話文「先日総理府が発表いたしました世論調査によりますと死刑を支持するという人が八十パーセント近くになっております」における節境界と係り受け構造の関係を図 2 に示す．ここで，節境界の判定は節境界解析 (CBAP)[3] により実行する．本来，文を節に一次元的に分割することは困難であるものの [2]，節境界解析により近似的に分割することは可能である [3]．本研究では，節境界解析により検出された節境界ではさまれた単位を**節境界単位**と呼び [2]，これを新たな解析単位と考える．係り受け解析は，入力された節境界単位の内部の係り受け構造を解析するとともに，既に入力された節境界単位の最終文節の係り先を可能であれば決定する．

本手法では，形態素解析，文節まとめ上げ，及び節境界解析が施された 1 独話を入力とする．ここで，入力データには，文境界が付与されていないことに注意されたい．また，この手法では，係り受けの後方修飾性，係り先の唯一性，非交差性の 3 つの性質を絶対的制約とする．解析の手順は以下の通りである．

1. **節レベルの係り受け解析**

   1 独話中のすべての節境界単位に対して，
   その内部の係り受け構造を解析する．

2. **独話レベルの係り受け解析**

   1 独話中のすべての節境界単位に対して，
   その最終文節の係り先を解析する．

上述した 2 つの解析とも，各文節間の係り受け確率を統計的に獲得し，それを用いて係り受け構造の

尤度を計算し，最尤の構造をもとめる．統計モデルの詳細は，文献 [5] を参照されたい．

ここで，節境界単位の最終文節の受け文節を同定する独話レベルの係り受け解析では，その受け文節がいつ入力されるかは明らかではないため，それを決定するタイミングが問題となる．本研究では，節境界単位が入力されるたびにその時点での最尤の係り受け構造を解析し，ある最終文節の係り受け関係が一定の入力回数（以下，固定値）変わらなかった場合，その受け文節を係り先として決定する．なお，この解析と同時的に要約単位の決定も行うため，具体的な解析アルゴリズムは次節で述べる．

3.2 要約単位決定アルゴリズム

独話レベルの漸進的係り受け解析に基づく要約単位決定の流れを以下に示す．独話レベルの係り受け解析では，節境界単位 C_i が入力されるごとに，すでに入力された節境界単位 $C_1 \cdots C_{i-1}$ の各最終文節 $b^1_{n_1} \cdots b^{i-1}_{n_{i-1}}$ に対する係り受け構造 $D = \{(dep(b^j_{n_j}), k) | \ 1 \leq j \leq i-1\}$ を更新することにより実行する．ここで k は $dep(b^j_{n_j})$ の不変回数を示す．以下に要約単位決定アルゴリズムを示す．なお，固定値を σ とする．

(1) 内部の係り受け構造が決定された節境界単位 C_i を入力する．(2) へ進む．

(2) 1 独話の文節列を $B(= B_1 \cdots B_m)$ とし，最終文節の係り先が未決定な節境界単位の集合 U を $U = \{C_h, \cdots, C_{i-1}\}(1 \leq h \leq i-1)$ とする．また，これらの最終文節を係り文節とするような係り受け構造 $\{dep(b^h_{n_h}), \cdots, dep(b^{i-1}_{n_{i-1}})\}$ を S_{last} とする．このとき，$P(S_{last}|B)$ を最大とする S_{last} をもとめる．(3) へ進む．

(3) (2) で同定された係り受け構造 S_{last} に基づき，最終文節に対する係り受け関係 D を更新する．ここで $dep(b^j_{n_j})(h \leq j \leq i-1)$ が同一の場合は不変回数を $k+1$ とし，異なる場合は 1 とする．(4) へ進む．

(4) $k = \sigma$ を満たす係り受け関係 $(dep(b^j_{n_j}), k) \in D$ に対して，文節 $b^j_{n_j}$ の係り先が決定したとして $dep(b^j_{n_j})$ を同定する．また，最終文節の係り先が決定された節境界単位を U から取り除く．(5) へ進む．

(5) (4) で U の中で最左の節 C_h の最終文節の係り先が同定された場合は，同定された係り受け関係によって連結される節境界単位の列を要約単位として決定し要約処理へ渡す．まだ節境界単位が入力される場合，(1) へ戻る．すべての節境界単位が入力された場合，(6) へ進む．

(6) $k < \sigma$ の $(dep(b^j_{n_j}), k) \in D$ に対して，その係り受け関係 $dep(b^j_{n_j})$ を同定する．要約単位として同定されていない残りの節境界単位の列を要約単位として決定し要約処理へ渡す．

3.3 要約単位決定の例

独話「正当な事由がない限り契約期間が切れたといっても明け渡しを請求できない点にあるといわれています」の節境界単位末の文節の係り先を解析する様子を図 3 に示す．(a)〜(f) の 6 つの過程から構成され，それぞれ上部に係り受け構造を，下部に節境界単位の最終文節の係り受け関係を示す．$(dep(b^j_{n_j}), k) \in D$ の $dep(b^j_{n_j})$ が係り文節及び受け文節に，k が不変回数に相当する．なお，ここでは固定値が 3 であるとして説明する．

(a) は，最初の節境界単位 I が入力された状態を，**(b)** は，節境界単位 II が入力され，係り受け構造 $\{dep($限り$)\}$ が解析された状態を示す．dep(限り) は上部の点線矢印に相当し，「限り」の係り先が「切れた」であり，不変回数は 1 であることが下部に記録される．同様にして，**(c)**，**(d)** は，それぞれ節境界単位 I II，IV が入力されたときの最尤の係り受け構造 $\{dep($限り$), dep($切れた$)\}$，$\{dep($限り$), dep($切れた$), dep($いっても$)\}$ が解析された状態を示す．

(e) は，節境界単位 V が新たに入力され，最尤の構造 $\{dep($限り$), dep($切れた$), dep($いっても$), dep($請求できない$)\}$ がもとまった状態を示している．このとき，係り受け関係 dep(切れた) の不変回数が 3 に達したため，この関係を決定し出力する．ここで，3.2 節の (5) が実行される．この場合，決定された係り受け関係 dep(切れた) を最終文節とする節境界単位 II は，最終文節の係り先が未決定な最左の節境界単位 I ではないため，要約単位はまだ同定されない．

(f) は，節境界単位 VI が新たに入力され，最尤の係り受け構造 $\{dep($限り$), dep($いっても$), dep($請求できない$), dep($あると$)\}$ がもとまった状態を示す．(e) と同様に不変回数が固定値に達している係り受け関係 dep(限り), dep(いっても) を決定し出力する．ここで，3.2 節の (5) が実行される．この場合，決定された係り受け関係 dep(限り) を最終文節とする節境界単位 I は，最終文節の係り先が未決定な節境界単位のうち，最左であるため，これまでに決定された係り受け関係で連結している節境界単位 I から IV までを要約単位として決定する．

4 係り受け構造に基づく要約

本手法では，前節で同定された要約単位に対して，冗長と思われる箇所を文節単位，節境界単位の順で削除する．そのとき，部分的な削除によって，不自然な日本語が生成されないように，係り受け構造を考慮して冗長箇所を削除する．

ここで，要約単位の文字列を何文字まで要約するのかという目標文字数の設定が問題となる．字幕生

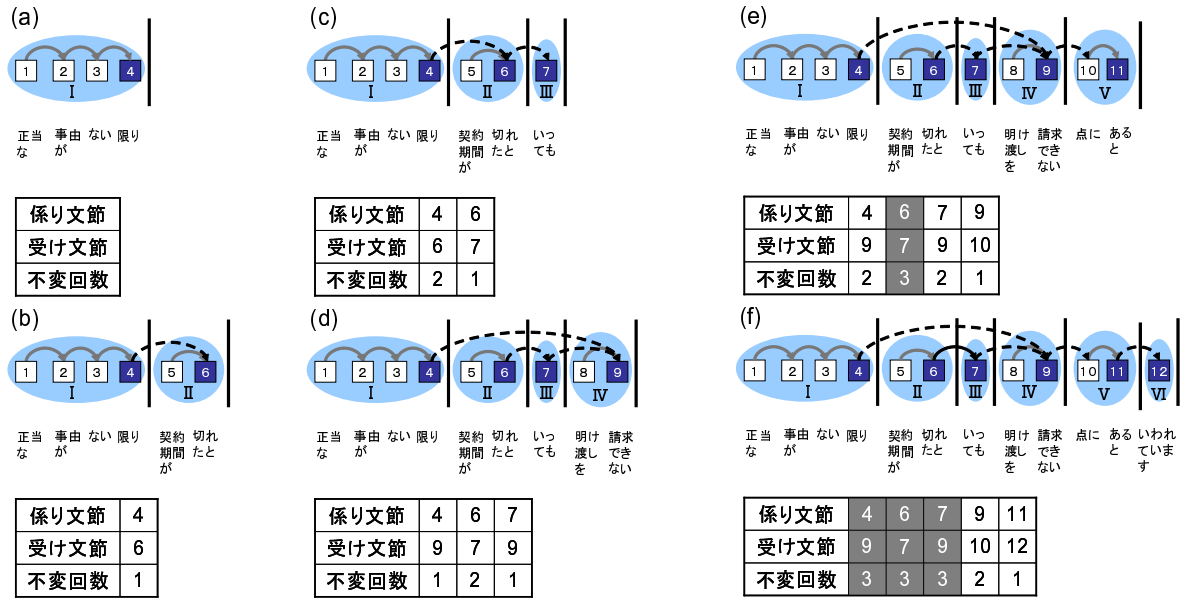

図 3: 要約単位決定の例 (固定値 3 の場合)

成では，話者の発話内容を分かり易く，かつ，字幕の表示時間内に聴衆が読みきれる程度に要約することがもとめられる．しかしその一方で，話者の発話音声をできる限りそのまま字幕化してほしいという要望もあり，そのバランスが問題となる．そこで，本研究では，聴衆が字幕テキストを読みきることができる字幕表示速度として，(株) 日本文字放送が字幕放送の作成において設定している「1 秒あたり 4 文字」という基準に着目した．この基準と 2 節で述べた字幕表示時間，すなわち，要約単位の発話時間から，要約する際の目標文字数を各要約単位ごとに設定した．

要約は，以下に示す (1)〜(3) の順で文節単位を削除し，続いて (4),(5) の順で節境界単位を削除する．また，各段階で複数の削除候補が見つかった場合は，節境界単位や要約単位の最終文節からの係り受けの深さによって順位を付け，深いものから順に削除する．ただし，目標文字数に達した時点で停止することとする．最終的に，条件 (5) を満たすものの削除が終わり，その時点でまだ目標文字数に達していない場合でも，発話内容の重要部分の欠落を避けるため，そのまま終了する．

4.1 文節単位の削除による要約

節末でなく，かつ，自分に係る文節がない文節について，以下の条件を満たすならば順に削除する．

(1) 自立語の品詞が副詞で，受け文節の自立語の品詞が副詞または形容詞であり，受け文節が節境界単位の最終文節でない．

(2) 受け文節の自立語の品詞が名詞（形式名詞以外）である．

(3) 受け文節が述語である．ここで，受け文節が述語であるか否かは，節境界単位の最終文節であるか否かで判断する．ただし，ラベルが “感動詞”，“体言止”，“談話標識”，“主題ハ” である節境界単位は除く．

4.2 節境界単位の削除による要約

要約単位の最後の節境界単位ではなく，かつ，自分に係る節境界単位がなく，かつ，節境界単位の最終文節の係り先が形式名詞ではないという条件を満たす節境界単位について，以下の条件を満たすならば順に削除する．

(4) 節境界単位に付与されたラベルが，従属度が低いため CBAP では検出されても CBAP-csj では検出されないような種類（連体節，ナガラ節，ツツ節など）（文献 [3] 参照）である．

(5) 節境界単位に付与されたラベルが，節境界直後の切れ目の大きさという観点から分類された 3 つのレベル「絶対境界」「強境界」「弱境界」のうち，「弱境界」（文献 [3] 参照）である．

5 評価実験

独話音声のリアルタイム字幕生成における本手法の有効性を評価するため，字幕生成実験を行った．

表 1: 要約品質の判断基準

評価値	判断基準
4	文章として自然に読むことが可能で（自然さ），かつ，重要箇所がすべて抽出されている（忠実度）.
3	自然さと忠実度が少しずつ損なわれている.
2	自然さと忠実度のどちらか一方が非常に損なわれているが，何とか理解可能である.
1	非常に不自然で読むことが難しく，重要箇所が頻繁に欠落しており，字幕テキストを読んでも理解できない.

表 2: 各番組の要約単位数，要約単位の文字数，要約率

	要約単位数	要約単位の文字数	要約率 (%)
番組 1	125	3,076	78.7
番組 2	107	2,877	78.2
番組 3	133	3,369	74.4
番組 4	109	2,997	79.6
番組 5	100	2,921	79.3
番組 6	123	3,263	75.4
番組 7	105	2,917	77.2

5.1 実験概要

実験には，NHK の解説番組「あすを読む」（番組あたりの長さは約 10 分）の書き起こしデータを使用した．書き起こしデータに形態素解析，文節まとめ上げを施した 7 番組 (470 文) に対して，字幕生成実験を行った．また，漸進的係り受け解析において，固定値は 3 とし，学習データは，形態素，文節，節，係り受けに関する情報が与えられた 95 番組 (5,532 文) を用いた．なお，これらの実験データは文献 [5] と同様のものである．

生成した字幕に対して，1）字幕表示速度（1 秒あたりの表示文字数），2）字幕テキストの品質を評価した．1）は，各要約単位を発話時間分だけそのままテキストを表示した場合（以下，書き起こし法）と比較した．また，2）は，削除が実行された各要約単位に対して，その書き起こしテキストと対応する字幕テキストを提示し，テキストの自然さ，及び，忠実度 [4] に基づいて，要約単位ごとに 4 段階で評価した．この判断基準を表 1 に示す．評価は二人で実施し，その平均点を各要約単位の評価値とした．

なお，字幕生成システムは GNU Common LISP で実装し，CPU が Pentium4 2.40GHz，メモリが 2GB の Linux PC 上で実験した．

5.2 実験結果

表 2 に番組ごとの要約単位数，要約単位の文字数，文字数に対する要約率（=字幕テキストの文字数/要約単位の文字数× 100）をそれぞれ示す．要約率は，全体で 77.4% (16,612/21,462) であった．また，要約単位は全体で 802 個同定され，文数の約 1.7 倍であった．

図 4 に，本手法と書き起こし法の，字幕表示速度ごとの字幕画面数の度数分布を示す．1 秒あたりの平均表示文字数は，本手法が 5.3(文字/秒) であるのに対して，書き起こし法は 6.5(文字/秒) であった．また，「1 秒あたり 4 文字」という基準を満たした字幕画面は，本手法が 28.4% (228/802)，書き起こし法が 2.0% (16/802) であった．これらから，本手法により，

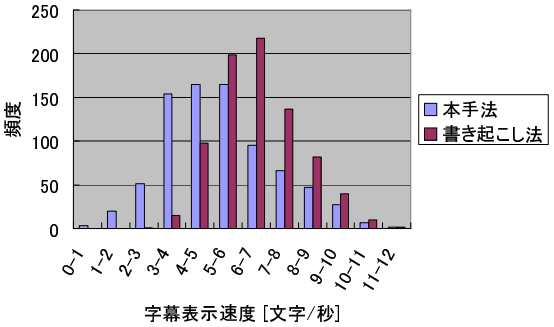

図 4: 字幕表示速度ごとの字幕画面数の度数分布

字幕の読み速度にゆとりが生じることがわかった．

次に，削除が実行された要約単位（551 個）に対して，字幕テキストの品質を 4 段階で評価した結果を図 5 に示す．この図は，各評価値が与えられた字幕画面の割合を示している．評価値 1 の要約単位の割合は 4.7%しかなく，良好な結果が得られている．一方，評価値 3 以上の要約単位は 303 個であり，全体の 55.0%を占めた．以上の結果から，本手法により，ほとんどの字幕が許容できる程度の品質を備えていることがわかった．

6 おわりに

本稿では，漸進的係り受け解析に基づくリアルタイム字幕生成手法を提案した．本手法では，節が検出されるたびに同定される係り受け構造に基づいて，要約処理を実行することにより，入力音声のリアルタイムな字幕化を実現している．実際の独話データを用いた実験により，本手法の実現可能性を確認した．

今後は，要約手法を洗練させ，要約の品質に関する評価を詳細に行う予定である．

謝辞 本研究は，総務省戦略的情報通信研究開発推進制度の研究委託「講演など独話データの知的構造化に関する研究開発」，ならびに，科学研究費補助

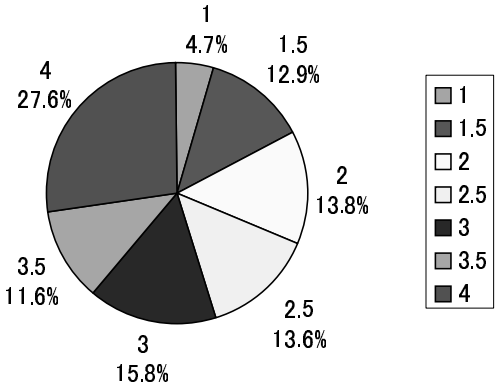

図 5: 要約品質評価値ごとの字幕画面数の度数分布

金（特別研究員奨励費）「大規模音声言語コーパスを用いた独話データの構造化とその応用に関する研究」（課題番号 18・6433）により実施したものである．

参考文献

[1] 堀智織，古井貞熙：単語抽出による音声要約文生成法，電子情報通信学会論文誌 D-II，Vol.J85-D-II，No.2，pp.200-209 (2002).

[2] 柏岡秀紀，丸山岳彦: 節境界単位による翻訳 - 連体節について -，言語処理学会第 10 回年次大会論文集，pp. 460–463 (2004).

[3] 丸山岳彦，柏岡秀紀，熊野正，田中英輝: 日本語節境界検出プログラム CBAP の開発と評価，自然言語処理，Vol. 11，No. 3，pp. 39–68 (2004).

[4] 三上真，増山繁，中川聖一：ニュース番組における字幕生成のための文内短縮による要約，自然言語処理，Vol. 6，No. 6，pp. 65–81 (1999).

[5] T. Ohno, S. Matsubara, H. Kashioka, N. Kato, and Y. Inagaki：Incremental Dependency Parsing of Japanese Spoken Monologue Based on Clause Boundaries, *Proc. of the 9th European Conference on Speech Communication and Technology,* pp. 3449–3452 (2005).