社団法人　情報処理学会　研究報告
IPSJ SIG Technical Reports

2005－ＳＬＰ－59（17）
2005／12／22

音声認識システムSOLONの日本語話し言葉コーパス（公開版Ver.1.0）による評価

中村 篤　大庭隆伸　渡部晋治　石塚健太郎　堀 貴明
マイク・シュスター　エリック・マクダーモット　南 泰浩

日本電信電話株式会社
NTT コミュニケーション科学基礎研究所 メディア情報研究部
信号処理研究グループ
〒619-0237 京都府相楽郡精華町光台 2-4
E-mail: {ats, oba, hori, ishizuka, watanabe, hori, schuster, mcd, minami}@cslab.kecl.ntt.co.jp

あらまし NTTコミュニケーション科学基礎研究所で開発を進めている音声認識研究用テストベッドSOLONの，日本語話し言葉コーパス(CSJ: Corpus of Spontaneous Japanese)公開版 Ver.1.0 を用いたベンチマーク評価結果を報告する．我々は実環境での自然な話し言葉を対象とした音声認識の研究を進めており，CSJを重要なベンチマークテスト用試料のひとつと位置づけている．本報告では，最小識別誤り学習や全共分散型モデルを始めとするいくつかの技法の大語彙自然発話音声認識における効果を実験により明らかにし，間投詞や精度算出言語単位の違いを考慮した追加評価等の結果もあわせて示す．

キーワード SOLON，重み付き有限状態トランスデューサ，最小識別誤り学習，全共分散型モデル

A Benchmark Evaluation of Speech Recognizer SOLON using The Corpus of Spontaneous Japanese (Ver.1.0)

Atsushi Nakamura, Takanobu Oba, Shinji Watanabe, Kentaro Ishizuka, Takaaki Hori,
Mike Schuster, Erik McDermott and Yasuhiro Minami

NTT Communication Science Laboratories
NTT Corporation, 2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0237

Abstract The SOLON is a speech recognition testbed system that has been developed at NTT Communication Science Laboratories. This paper reports results from the latest benchmark evaluation of the SOLON using the Corpus of Spontaneous Japanese (CSJ). The effectiveness of some of techniques, including minimum classification error training and full-covariance modeling, is presented through experiments. Also, results of recognition error analysis and additional evaluations are described.

Keyword SOLON, Weighted Finite-state Transducer, Minimum Classification Error Training, Full-covariance Modeling

1. はじめに

NTT コミュニケーション科学基礎研究所では音声認識研究のためのテストベッドとして音声認識システムSOLONの開発を進めている．SOLONには重み付き有限状態トランスデューサ(WFST: Weighted Finite-state Transducer) の高速on-the-fly合成による効率的認識アルゴリズム[1]を中心として，最小識別誤り(MCE: Minimum Classification Error)学習[2,3]，ベイズ的音声認識(VBEC: Variational Bayesian Estimation and Clustering for speech recognition)[4]等の独自技術が実装されている．

実環境での自然発話を対象とした音声認識の実現を目指す上で，その試料となる音声データベースの役割が重要であることはいうまでもない．日本語話し言葉コーパス(CSJ: Corpus of Spontaneous Japanese)は，現実の学会講演などを主たる対象とし，約660時間もの音声データを収録した大規模な自然発話音声データベースであり，音声言語処理にとどまらない幅広い領域での有効利用が期待されている[5,6]．我々は当初より，このCSJを重要な音声認識ベンチマーク評価用試料のひとつと位置づけており，既に2001，2002年にかけて配布されたモニター版を用いSOLONの評価を実施してきた[7]．本稿では2004年6月に配布が開始された「CSJ公開版 Ver.1.0」を用いて新たに実施したベンチマーク評価の結果を報告する．

SOLONに実装された独自技術の中でも特にMCE学習に関し，近年連続音声認識タスクでの評価を本格化させており[8]，今回CSJを対象として，その大語彙自然発話音声認識における効果を検証する．これと並行し，今回新たに実装した全共分散パラメータの最大尤度推定の効果も確認する．また言語モデルに関して，語彙拡大の効果を評価するほか，backoffによるN-gram確率平滑化の際にKneser-Ney法[9]を採用し，既に広く用いられているWitten-Bell法との比較を行う．さらに，認識誤り傾向の分析結果を受け，自然発話音声に

特徴的な間投詞の取り扱いや，正解精度算出時の言語単位の相違を考慮して実施した追加評価の結果等もあわせて示す．

2. SOLON ベンチマーク評価の基本設定

ベンチマーク評価にあたり，まずSOLONに対し種々の技法を適用する前の実験条件と認識アルゴリズムについて述べ，さらにそれらに基づくベースライン認識性能を定める．

2.1. 実験条件

表 1〜4 に実験条件を示す．CSJ モニター版による評価[7]時から特徴量ベクトルに変更(25 次元→39 次元)が加えられたが，総じて標準的な実験条件といえる．評価データは CSJ 公開版で規定されている「男性話者による学会 10 講演(test-set-1)」[6]とし，事前に発話単位への分割等を行うことなく，講演 1 回分全体をそのまま認識対象とする．性能は CSJ 公開版の短単位による単語正解精度で評価する．学習データとしては，音響モデル用に評価データを除く男・女性話者による学会講演を，言語モデル用に評価データを除く学会・模擬の全講演を用いた．

2.2. 認識アルゴリズム

SOLON では WFST による探索空間表現に基づく音声認識アルゴリズムを採用している．WFST は入力記号列に従って状態遷移を繰り返し，その状態遷移過程に割り当てられた出力記号列を出力する汎用的な記号列変換の計算モデルである．状態遷移には重みが付与されており，これを問題に応じてコスト，確率等に対応づける．音声認識の場合，重みを対数尤度の反数に対応させることで，最小重みつまり最大尤度を基準に記号列変換を行う．SOLON では，以下の 4 レベルの WFST を構成し，これらを統合して得られる単一の WFST を用いた One-pass Viterbi decoding により音声認識を行うことができる．

H: HMM の状態列から triphone 列 への変換
C: triphone 列から音素列への変換
L: 音素列から単語列への変換
G: 単語列から単語列(trigram)の変換

WFST を一つに統合するには，WFST の基本操作である，合成(composition)，決定化(determinization)，最小化(minimization)等を用いる．WFST の合成により，複数の WFST が任意の記号列を順に変換するのと等価な変換を実現する一つの WFST を作ることができる．今 WFST の合成を二項演算子"∘"で表すとしたとき，合成 $H \circ C \circ L \circ G$ の結果は HMM の状態系列を単語 trigram 上の単語列に変換する WFST となる．さらに，決定化や最小化により，この WFST を最適化することで効率的な音声認識が実現される．

ところが現実問題として，音声認識で用いるすべての知識を単一の WFST に統合しようとすれば保持すべき状態や状態遷移の数は組合せ的に増加し，認識時の所要メモリ量が膨大なものとなってしまう．これを回避する手段として on-the-fly 合成がある．WFST の on-the-fly 合成に基づく音声認識では，事前に合成しておく WFST と認識処理時に必要に応じて合成する WFST を用意し，これらを認識中に逐次合成する．本報告では次式のように HMM 状態列を単語列(unigram)に変換する WFST と，単語列(unigram)を単語列(trigram)に変換する WFST に分解している．

$$(H \circ C \circ L_{uni}) \circ G_{tri\,/\,uni} \tag{1}$$

ここで，L_{uni} は単語辞書に unigram 確率に基づく重みを付与した WFST であり，$G_{tri\,/\,uni}$ には L_{uni} に付与された重みを補正するために trigram 確率を unigram 確率で割った値に基づく重みが付与されている．このように G の重みを分解して保持することは，on-the-fly 合成において言語モデルの知識を探索の早い段階で利用するための先読み効果をもたらす．

さらに我々は，WFST の on-the-fly 合成を効率的に行うためのアルゴリズムを提案している[1]．今，二つの WFST A と B を on-the-fly 合成することを考える．入力記号列 X が与えられたとき，A と B による記号列の変換は，

表 1. 音声分析

標本化周波数/ 量子化精度	16kHz/16bit
特徴量ベクトル (39 次元)	(12 次 MFCC+ 対数パワー) + Δ + Δ²
分析フレーム長/周期	25ms/10ms (Hamming)
特徴量正規化	ケプストラム平均正規化

表 2. 音響モデル

音響モデル種別	3 状態 left-to-right HMM 対角型共分散行列
音素カテゴリ数	43
音素決定木質問数	144

表 3. 言語モデル

言語モデル種別	単語 trigram 確率モデル
語彙数	30,000 (30k)
N-gram 確率平滑化	Backoff 平滑化 (Witten-Bell 法)
N-gram カットオフ	bigram:1, trigram: 1

表 4. 学習／評価データ（全て CSJ 公開版より）

音響モデル 学習データ	学会 961 講演 234 時間
言語モデル 学習データ	学会及び模擬 2,672 講演 異なり単語数 100,807 延べ単語数 6,830,031
評価データ	男性話者学会 10 講演 延べ単語数 26,329 パープレキシティ 70.1 OOV 率 2.3%

$$W(X \to \hat{Z}) = \min_{Z} W_{A \circ B}(X \to Z) \quad (2)$$

$$= \min_{Y,Z}(W_A(X \to Y) + W_B(Y \to Z)) \quad (3)$$

を与える記号列 $\hat{Z}$ を求めることである．ここで，$W_M(\alpha \to \beta)$ は，WFST M を用いて記号列 α を β に変換するときの累積重みである．従って，$\hat{Z}$ を求めることは，累積重み最小となる記号列を探索する問題となる．高速 on-the-fly 合成では，式(3)を次のように書き換える．

$$W(X \to \hat{Z}) = \min_{Y}(W_A(X \to Y) + \min_{Z} W_B(Y \to Z)) \quad (4)$$

この式で，$\min_Z W_B(Y \to Z)$ を $W_A(X \to Y)$ の補正項と捉えると，基本的には WFST A のみを用いて $\hat{Y}$ を求めるアルゴリズムを $W(X \to \hat{Z})$ の計算においてそのまま適用できることになる．

高速 on-the-fly 合成による記号列変換では，WFST A に基づいて Viterbi decoding を行い，その過程で生成される部分パス仮説の中間累積重みを式(4)のように計算する．補正項は部分パス仮説の出力記号列が変化したときのみ WFST B にアクセスして計算する．入力記号列 X に従い最後まで処理を進め，$\hat{Y}$ が求まった時点で $\min_Z W_B(\hat{Y} \to Z)$ となる Z を出力結果 $\hat{Z}$ とする．実際には，$\hat{Z}$ は補正項を計算する際に求まっており，その結果を利用することができる．特に，A の出力記号に ϵ (何も出力しないことを表す記号)が多い場合は，A から出力される記号列が変化する頻度が少なくなり，補正項の計算回数は極小化される．式(1)において WFST A に相当する $H \circ C \circ L_{uni}$ の出力記号の多くは ϵ であることから，この分解($H \circ C \circ L_{uni}$ と $G_{tri/uni}$)はまさに高速 on-the-fly 合成向きになっているといえる．

以降の実験では全て，この WFST の高速 on-the-fly 合成に基づく認識アルゴリズムを用いている．

2.3. ベースライン認識性能

ここまで述べた条件の下でのベースライン性能を定めるため，音響モデルの構造を変化させ各々の構造について単語正解精度を算出した．ビーム幅は精度が十分飽和する値を用いている．表 5 に，音響モデルの構造ごとの単語正解精度を示す．表中のケプストラム平均正規化(CMN: Cepstral Mean Normalization)におけるケプストラム除去量は，以降の実験設定に合わせ，発話単位，講演単位の二種類の方法によって計算している．発話単位の計算では認識時(講演単位)との不整合が生じるため，講演単位の計算に比べ若干精度が低くなるが，それが必ずしも両者の優劣を示すものではないことに注意されたい．結果として，状態数 5,000，状態当りガウス分布混合数 32 の条件で最も高い精度 78.2%が得られている．以下，これらをベースライン性能とし，必要に応じて参照する．

表 5. ベースライン性能（単語正解精度(%)）

状態数＼混合数	8		16		32	
1000	-	74.3	-	75.5	-	76.3
2000	-	75.9	76.6	76.9	77.6	77.3
3000	75.9	76.3	76.9	77.2	-	78.0
4000	-	76.6	77.2	77.4	-	77.9
5000	-	76.6	-	77.2	-	**78.2**

各欄において CMN のケプストラム除去量を，左:発話単位で計算した場合(発話単位 CMN)，右:講演単位で計算した場合(講演単位 CMN)，認識はいずれも講演単位

3. 諸技法の実装と効果

全共分散パラメータの導入や認識語彙の拡張等によって性能の向上を試み，その効果を検証する．

3.1. 全共分散パラメータの最大尤度推定

多次元ガウス分布により特徴量ベクトルの出力分布を表現する HMM において，共分散パラメータを成分とする行列(共分散行列)として，対角成分のみを許す対角共分散行列が広く用いられてきた．非対角成分を許す共分散行列(対角共分散行列に対し，全共分散行列と呼ばれる)は特徴量間の相関を考慮した，より精密な表現を可能とするものの，モデルパラメータ数が膨大となるため過学習を起こしやすいという実用上の問題があった．近年，学習データの量が格段に大きくなったことで，全共分散行列の各成分(全共分散パラメータ)をバランス良く最大尤度推定することも可能となってきている[10]．全共分散パラメータの推定において必要とされる行列計算は，共分散行列の三角化，逆行列計算，及びガウス分布規格化定数のための行列式計算である．全共分散パラメータは実対称正定値行列で表現されるため，それに特化した三角化アルゴリズムとして Cholesky 分解が用いられる．三角化された行列により，逆行列及び行列式を効率よく計算することができる(例えば [11])．

表 6 に，得られた全共分散型モデルの性能を示す．総パラメータ数の関係で、対角共分散パラメータの場合(表 5 参照)と比べて，小さい状態数，混合数のモデル構造が高い単語正解精度を与える傾向にある．同じモデル構造同士で比較するとその全てで一貫して精度向上(1.5～3.3 ポイント)が見られる．結果的に得られた最良精度 79.8%(2000 状態×16 混合)は，ベースラインでの最良精度 78.2%(5000 状態×32 混合)と比べて 1.6 ポイントの向上となった．

3.2. 認識語彙の拡張

これまで我々は，本報告でもベースラインとしたように，言語モデルとして語彙数 30,000(30k)の単語 trigram 確率モデルを標準としてきた．ここでは，語彙を 65,000 語(65k)，さらには学習データに出現する全単語(約 10 万語，100k)へと拡張した場合の効果を評価す

表 6. 全共分散型モデルの性能

状態数＼混合数	8		16		32	
1000	-	77.7	-	78.7	-	78.9
2000	-	79.0	79.6	**79.8**	79.5	79.4
3000	-	79.4	79.7	79.5	-	78.8

各欄，左:発話単位 CMN，右:講演単位 CMN
認識はいずれも講演単位

表 7. 語彙拡張の効果(対角共分散型モデル)

状態×混合数	W-B	修正K-N		
	30k (2.28)	30k (2.28)	65k (1.68)	100k (1.36)
5000×32	78.2	78.2	78.4	**78.6**

W-B 30k の数値は表 4 より再掲，括弧内:OOV 率(%)

表 8. 語彙拡張の効果(全共分散型モデル)

状態×混合数	W-B	修正K-N		
	30k (2.28)	30k (2.28)	65k (1.68)	100k (1.36)
2000×16	79.8	79.9	80.2	**80.3**

W-B 30k の数値は表 5 より再掲，括弧内:OOV 率(%)

表 9. 修正 Kneser-Ney 法と原型 Kneser-Ney 法

状態×混合数	修正K-N	原型K-N
2000×16	80.3	80.4 → **80.5**

る．また，backoff による trigram 確率の平滑化において従来用いてきた Witten-Bell(W-B)法に代わり，Kneser-Ney(K-N)法[9]を導入する．

従来の backoff では学習データ中で観測されない N-gram の確率を(N-1)-gram 確率(に比例する値)でそのまま代用するため，出現頻度が高く，かつその値が先行単語(列)に強く依存する単語に対し推定値が悪くなるという問題がある．例えば"円"という単語は数字の後に多く出現するが，その頻度自体が大きいために，先行単語が何であれ unigram に backoff した場合は高い確率となってしまう．一方 K-N 法では，対象とする N-gram が存在する場合と存在しない場合で(N-1)-gram の分布が異なることを考慮し N-gram としては観測されない(N-1)-gram のみから計算された平均(N-1)-gram 確率(に比例する値)を用い，absolute discounting に基づいて最終的な代用 N-gram 確率を決定する．これにより上記"円"のような単語の確率が過剰に高くなるのを抑えることができる．K-N 法において absolute discounting による補正をより精密に定義することにより導出された修正 K-N 法が提案され，さらに高い精度での N-gram 確率推定が可能と報告されている[12]．そこで本報告ではまず，修正 K-N 法を導入した．

表 7，8 に 30k，65k，100k の単語 trigram 確率モデルを用いた場合の単語正解精度を示す．対角共分散型，全共分散型いずれの場合も OOV 率の軽減に伴い少しずつ精度が上がっていき，語彙 100k において対角共分型散モデル 78.6%，全共分散型モデル 80.3%の精度を得た．修正 K-N 法は全共分散型モデルの語彙 30k において僅かに W-B 法を上回っており，パープレキシティも全ての語彙で修正 K-N 法の方が低い(例えば，語彙 30k で 70.6→67.7)．一方，原型の K-N 法との比較では，原型 K-N 法の方がやや低いパープレキシティを与えた(例えば，語彙 100k で 76.2→75.3)．そこで，全共分散型モデルの語彙 100k について，原型 K-N 法による言語モデルを用いた再評価を試みたところ表 7 の結果を僅かに上回り(80.4%)，修正 K-N 法の原型 K-N 法に対する優位性は今回示されなかった．さらにビーム幅を微調整することで，最終的な単語正解精度は 80.5%となった(表 9)．

4. 最小識別誤り学習

MCE 学習そのものについては既に多くの文献が存在するので詳細はそれらを参照されたい[2,3]．特にここで報告するのは WFST に基づく大語彙 MCE 学習[8]の効果である．

4.1. WFST に基づく大語彙 MCE 学習

MCE 学習を大語彙連続音声認識に適用する場合，損失値の計算に必要な対立仮説を得るために学習時にも言語モデルを用いた音声認識を動作させる．本報告ではいずれも語彙 100k の単語 unigram 確率モデル，単語 trigram 確率モデル(以下 unigram モデル，trigram モデル)の二種類を用いた．unigram モデルを用いる場合，同音異義語同士の区別がなくなるためそれらを統合すると実効的な語彙数は 68,000(図中 68k)程度となる．これらの言語モデルは WFST で表現され，図 1 に示す MCE 学習の流れの中で利用される．さらに，正解単語列もまた，triphone 列を単語列に変換する WFST として扱われている．損失値の積算により得られた総損失は，学習データ全体についての認識誤り発生の度合いに対応している．総損失の具体的意味づけは対立仮説と正解単語列の尤度差を損失値に変換する損失関数の選び方によって異なる．本報告では線形の損失関数を用いており，総損失は，個々の誤りがもたらす，学習により解消すべき尤度差の総和に対応する．MCE 学習ではこの総損失を最適化手法によって最小化(理想的には負値に)するよう学習を進める．最適化手法としては，unigram モデルの場合は Quickprop 法を，trigram モデルの場合は Rprop 法を用いた[13]．

なお，本報告の MCE 学習においては対角共分散型モデルのみを評価の対象とした．また学習時は ML，MCE ともに CMN の除去量を発話単位に計算している(つまり，ベースラインは表 5 の左欄)．これは本実験の設定都合上の措置であるが，結果的に初期モデルの段階から学習と認識(講演単位 CMN)の不整合を生み，本実験に不利な条件を課すこととなっている．

4.2. 認識実験結果

まず表 10 に語彙 30k における単語正解精度を示す．MCE 学習時は全て unigram モデルを用いている．5 種

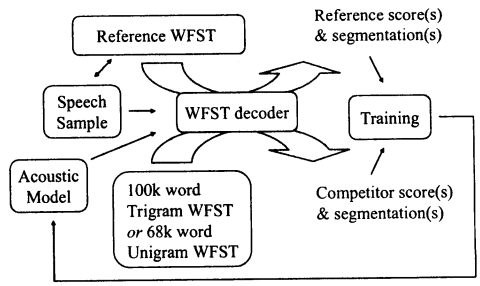

図 1. WFST に基づく大語彙 MCE 学習の流れ

表 10. 大語彙 MCE 学習の効果(認識語彙 30k)

状態×混合数	MCE学習時言語モデル	ML	MCE
2000×16	unigram	76.6 (23.4)	77.7 (22.3)
2000×32	unigram	77.6 (22.4)	79.0 (21.0)
3000×8	unigram	75.9 (24.1)	77.5 (22.5)
3000×16	unigram	76.9 (23.1)	**79.2 (20.8)**
4000×16	unigram	77.2 (22.8)	**79.2 (20.8)**

対角共分散型モデルでの単語正解精度(括弧内誤り率),
ML の数値は表 4 から再掲

表 11. 大語彙 MCE 学習の効果(認識語彙 100k)

状態×混合数	MCE学習時言語モデル	ML	MCE
2000×16	unigram	77.0 (23.0)	78.9 (21.1)
3000×16	unigram	77.6 (22.4)	79.5 (20.5)
4000×16	unigram	77.9 (22.1)	**79.9 (20.1)**
2000×16	trigram	77.0 (23.0)	**79.8 (20.2)**

対角共分散型モデルでの単語正解精度(括弧内誤り率)
ML の数値は発話単位 CMN にて別途算出

類の HMM 構造全てについて精度向上(1.1～2.3 ポイント)が見られる．3000 状態，16 混合のモデル構造で得られた 79.2%は，同じ状態数，混合数の共分散型モデルによる正解精度(79.7%，表 6)と比べても遜色ない．

続いて語彙 100k における単語正解精度を表 11 に示す．unigram モデルで MCE 学習を行った 3 種(2000/3000/4000 状態×16 混合)と，trigram モデルで MCE 学習を行った 1 種(2000 状態×16 混合)の 1 種を対象とした．ML，MCE ともに，認識時には修正 Kneser-Ney 法による backoff 平滑化を行った言語モデルを用いている．unigram モデルによる学習の効果により，3 種のモデルについてほぼ等しく 2 ポイント程度の精度向上がもたらされた．本実験で最良の正解精度である 79.9%(4000 状態×16 混合)は，同じ言語モデル(語彙 100k，修正 Kneser-Ney 法)で共分散型モデルが達成した 80.3%(表 8)に匹敵する．さらに trigram モデルによる学習はこの半分の規模のモデル構造(2000 状態×16 混合)で同等の正解精度(79.8%)を達成している．この結果は，前述の通り CMN 計算単位の学習－認識間不整合なる不利な条件下にもかかわらず，MCE 学習による対角共分散型モデルがパラメータ数 10 倍の全共分散型モデルと十分対等に渡り合えることを示している．

5. 評価結果の分析と考察

5.1. 性能向上と各技法の関係

3, 4 節で示したベースラインからの段階的な性能向上とそのために用いた技法との関係を図 2 に整理した．各ノードに付した値は当該条件下でモデル構造も考慮して暫定的に得られている最良の単語正解精度(全て再掲)である．各ノードに先行するノードでの精度がアークに付された技法を用いるときの初期値では必ずしもないことに注意されたい．この図から全共分散パラメータ，MCE 学習の効果の大きさがわかる．また，語彙の拡張も一貫して性能向上に寄与していることが見てとれる．

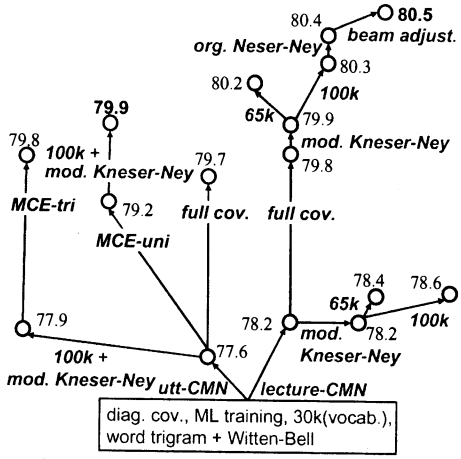

図 2. 性能向上における各技法の寄与

5.2. 認識誤りの傾向

全体で最良の単語正解精度(80.5%)をもたらした実験の認識結果テキストから大まかな誤りの傾向を分析する．まず単語単位で見ると，正解単語列中で置換または脱落した単語の件数順上位 11 位までは，助詞を中心とする一拍語であった．この 11 種の合計だけで 1250 件となり，全誤り(5144 件)の約 24%を占める．また全誤りの 6 割以上(3270 件)は置換誤りとして集計されており，その具体的事例の件数順上位には，助詞等の置換に混じり，間投詞同士の置換が目立つ．特に"え"と"えー"(140 件)，"ま"と"まー"(55 件)等，書き起こし上の表記ゆれとも見なせるものがかなりの数にのぼる．続いて音素単位では，置換誤り(3529 件)と脱落誤り(3235 件)の合計で全誤り(8162 件)の 8 割以上となる．特に同種の長短母音の間で 839 件もの置換誤りが起きており，これだけで置換誤りの約 24%，全誤りの 1 割以上を占める．

発話速度変動が大きい自然発話音声の認識において助詞等短い単語の誤りがさらに顕著となることは以前より知られており，今回の評価結果はあらためてそれを裏付けている．長短母音の置換誤りについては各単語発音の複数化等である程度対処できる可能性はあるが，抜本的な解決のためには継続長モデル等の導入必要性も示唆される．特に日本語は長短母音の区別を単語レベルの意味弁別に用いる言語であり，正確な長短母音認識の必要性は高い．ただし，上述の間投詞の例("え"と"えー")のように必ずしも完全な表層形の同定を必要としない場合においては，性能評価のありかたにも検討の余地があるものと考える．

5.3. 性能評価に関する考察

まず間投詞(ここでは便宜的に言い淀みも含める)の扱いについて考える．上の例("え"と"えー")のような単語末が長音か否かという程度の相違であれば，まずは，両者を区別せず同じ単語として扱い，その上で算出した単語正解精度を実効的な性能と捉えることが考

えられる．しかしながら，"ええ"，"えっ"，"えーえ"等，同様に区別の要否を問われる単語は数多く存在し，その判断から恣意性を排除することは容易でない．そこでここでは，(1)CSJ公開版の短単位を全てそのまま用いる従来の単語正解精度，に加え，(2)間投詞を全て同一の単語と考えて算出した単語正解精度(間投詞単一化)，(3)正解単語列，認識結果単語列の双方から間投詞を除いて算出した単語正解精度(間投詞除外)，の三種類の評価値を示す．(2)は，間投詞の出現位置が正しく認識できれば表層形によらず正解とするもので，間投詞の表層形が発話の意味内容に影響を与えることはないものの，当該位置に間投詞が現れたこと自体は認識すべきという立場である．(3)は，正解と認識結果の双方から間投詞を除外して評価するため，ある単語が間投詞への置換誤りを起こしていた場合は当該単語の脱落誤りとして数える．また間投詞が間投詞以外の単語に置換誤りを起こしていた場合は当該単語の挿入誤りとして数える．そして間投詞自体の脱落，挿入誤りは無視することとなる．これは間投詞を語彙外単語として扱うもので，単語スポッティングの考え方に近い．算出結果を表12に示す．(2)，(3)では精度が82%以上となっており，認識誤りの1割程度は間投詞に閉じたものであることがわかる．

次に正解精度算出の言語単位について考える．本報告では一貫してCSJ公開版の短単位をもとに言語モデルの学習と単語正解精度の算出を行っているが，本来単語の定義が明確でない日本語においては，認識向きに単語相当の言語単位を適切に定義すること自体も性能に有意な影響を与える重要な課題となる．しかしながら，異なる任意の単語定義に基づいて算出された単語正解精度同士は単純比較できず，性能の優劣を決めるのは困難である．そこで，言語モデルの単位として定義された単語の正解精度に加え，認識結果単語列からの再集計により書き起こし表記上の最小単位である仮名漢字まじり文字(仮名漢字)をもとにした正解精度を算出する．これにより異なる単語定義に基づく評価間でもある程度の性能比較が可能となる．表13に算出結果を示す．単位が細かくなると数値上の精度が上がり，仮名漢字単位では83.6%となった．さらに漢字部分をよみがなに置き換えたもの(仮名)や最小言語単位である音素に基づく精度も合わせて算出したところ，仮名単位で88.2%，音素単位では90%を超えた．これらは，同音異義語間の置換等を中心とした，現状の言語モデルだけでは対処の困難な誤りが相当件数存在していることを示唆している．

表12. 間投詞の扱いを考慮した評価

(1)単語正解精度	(2)間投詞単一化	(3)間投詞除去
80.5	82.2	82.9

表13. 種々の言語単位に基づく正解精度

単語(CSJ短単位)	仮名漢字	仮名	音素
80.5	83.6	88.2	90.5

6. まとめ

本稿では音声認識システムSOLONの，日本語話し言葉コーパス(CSJ)公開版 Ver.1.0 を用いたベンチマーク評価の結果を報告した．語彙100kを用いた最終的な単語正解精度は全共分散型モデルで80.5%となった．対角共分散型モデルでもMCE学習により79.9%という全共分散型モデルに匹敵する精度が得られた，今後，話者適応や，全共分散モデルとMCE学習の組合せ等によりさらなる性能向上が期待できる．

謝辞「日本語話し言葉コーパス」の構築に尽力されたすべての皆様に敬意と謝意を表します．

文　献

[1] T. Hori, C. Hori and Y. Minami, "Fast on-the-fly composition for weighted finite-state transducers in 1.8 million-word vocabulary continuous-speech recognition," in Proc. Interspeech'04 (ICSLP'04), Vol. I, pp. 289–292, 2004.

[2] E. McDermott, "Discriminative training for speech recognition," Ph.D. thesis, Waseda University, School of Engineering, March 1997.

[3] S. Katagiri, C-H. Lee, and B.-H. Juang, "New discriminative training algorithms based on the generalized descent method," in Proc. NNSP'91, pp. 299–308, 1991.

[4] S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda, "variational Bayesian estimation and clustering for speech recognition," IEEE trans. SAP, vol.12, pp.365-381, 2004.

[5] T. Kawahara, H. Nanjo and T. Shinozaki, and S. Furui, "Benchmark test for speech recognition using the corpus of spontaneous Japanese," in Proc. SSPR'03, pp. 135-138, 2003.

[6] "日本語話し言葉コーパス," 国立国語研究所 web page. (http://www.kokken.go.jp/katsudo/kenkyu_jyo/corpus/)

[7] 堀貴明，渡部晋治，マクダーモットエリック，南泰浩，中村篤，"音声認識システムSOLONの日本語話し言葉コーパスによる評価," 第3回 話し言葉の科学と工学ワークショップ，pp.85-91, 2004.

[8] E. McDermott and S. Katagiri, "Minimum classification error for large scale speech recognition tasks using weighted finite state transducers," in Proc. ICASSP'05, vol. 1, pp. 113–116, 2005.

[9] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in Proc. ICASSP'95, vol. 1, pp. 181-184, 1995.

[10] M. Schuster, T. Hori and A. Nakamura, "Mixtures of probabilistic principal component analyzers in speech recognition," 信学技法，SP2004-92，pp.67-71, 2004.

[11] W. H. Press B. P. Flannery, S. A. Teukolsky and W. T. Vetterling (丹慶勝市等訳), "Numerical recipes in C [日本語訳]," 技術評論社, 1993.

[12] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," Technical Report TR-10-98, Computer Science Group, Harvard University, 1998.

[13] J. Leroux and E. McDermott, "Optimization methods for discriminative training," in Proc, Eurospeech'05, pp. 3341-3344, 2005.