発音推定のための重みベクトルの適応的正則化手法における ハイパーパラメータの改善*

☆久保慶伍, サクティサクリアニ，ニュービッググラム，戸田智基，中村哲 (奈良先端大)

1 はじめに

様々な単語を認識・発話することが求められるオープンドメインな音声認識や音声合成において，辞書になく発音が分からない単語（未知語）の問題は重要な課題である．単語の発音が分からない場合，その単語を認識・発話することができないからである．この世に存在する全ての単語に対して人手により発音を付与するには非現実的な時間やコストが掛かるため，この問題を解決するために発音推定が行われる．オープンドメイン化や多言語化が進む音声認識分野では，その必要性がますます増加している．

発音推定は書記素列（Graphemes）から音素列（Phonemes）へと変換する g2p 変換で行われる．g2p 変換の分野ではこれまでに結合系列モデル [1] などの機械学習に基づく様々な手法が提案されてきた．最近の試みとして豊富な特徴量を用いるオンライン識別学習がある [2, 3, 4]．その代表的な手法として Margin Infused Relaxed Algorithm (MIRA) [5] を g2p 変換に適用するために構造学習に拡張した手法 [2, 3] がある．構造学習とは分類するクラス（例えば発音）が部分クラス（例えば音素）の組み合わせからなる分類問題を取り扱う機械学習の分野の一つである．

我々も二値分類手法である重みベクトルの適応的正則化手法（AROW: Adaptive Regularization of Weight Vectors）[6] を g2p 変換用に拡張した構造化 AROW を提案した [4]．これは 2 次統計量により表される各重みの現在の値に関する信頼度を用いることで，MIRA が持つ過学習問題を解決するオンライン識別学習法である．しかしながら，構造化 AROW はまだ完全な手法ではない．構造化 AROW の問題点として各重みの信頼度が早期の段階で高くなりやすいことが挙げられる．信頼度の逆数は直感的に言えばその重みの学習率を意味しているため，学習の早期の段階で信頼度が高くなりすぎると重みが不適切な値に収束し，それ以後の学習データにおいてその重みを動かせないという問題を引き起こしやすい．この問題は構造化 AROW の識別能力を低下させる．

上記の問題は二値分類の AROW においても存在する．二値分類では上記の問題を解決するために，Narrow AROW (NAROW) [7] と呼ばれるオンライン識別学習法が提案されている．この手法は AROW のハイパーパラメータを信頼度が高くなり過ぎないように設定することで上記の問題を防ぐ．また，その設定は NAROW の分類に関する誤り限界を最小化するように導出されるため，解析的な観点からみても，それは AROW の識別能力を改善するということが分かる．本論文では，この手法を g2p 変換に適用するために，NAROW を g2p 変換用に拡張した構造化 NAROW を提案する．さらに様々な g2p 変換タスクにより構造化 NAROW を評価する．

2 既存の g2p 変換手法

まず，本論文で用いる線形識別器による g2p 変換を以下のように定義する．

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \boldsymbol{w}^{\mathrm{T}} \Phi(\boldsymbol{x}, \boldsymbol{y}) \tag{1}$$

ここで，$\boldsymbol{x}$ は書記素列，$\boldsymbol{y}$ は音素列を表す．また，$\boldsymbol{w}$ は特徴量の重みベクトルを表し，$\Phi(\boldsymbol{x}, \boldsymbol{y})$ は，$\boldsymbol{x}$ と $\boldsymbol{y}$ の結合 N-gram の頻度などからなる特徴量ベクトルを表している．式（1）の $\hat{\boldsymbol{y}}$ はビーム探索などの探索アルゴリズムにより得られる．

上記の g2p 変換器において正確に発音を推定する重みベクトル $\boldsymbol{w}$ を得るために，オンライン識別学習を用いる．g2p 変換は構造学習問題であるため，構造学習に対応したオンライン識別学習を用いる必要がある．構造学習問題は，事前に有限個のクラスを定義する多値分類問題と異なり，部分クラスの組み合わせによりクラスが決定するためクラス数が無数にある．それ以外は基本的に多値分類と同じであるため，更新時に考慮するクラスの数を N-best などに限定する多値分類の学習手法は構造学習にも素直に適用できる．この後説明する手法はそれに該当する手法である．

g2p 変換において現在最も高い性能を誇るオンライン識別学習は MIRA に基づく手法である．i 番目のデータ $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ と現在の重み $\boldsymbol{w}_{t-1}$ により推定されたそのデータの発音の N-best 仮説 $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$ が与えられた時，MIRA は以下の制約付き最適化問題を解くことにより次の重みベクトル $\boldsymbol{w}_t$ を得る．

$$\arg\min_{\boldsymbol{w}_t} \frac{1}{2}\|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|^2;\ \text{s.t. } \forall n;\ \ell(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{w}_t) = 0 \tag{2}$$

ここで ℓ は正しい発音 $\boldsymbol{y}_t$ のスコアが十分な差で仮説の発音 $\hat{\boldsymbol{y}}_t$ よりも高い時に 0 の値を出力し，それ以外の時に正の値を出力する損失関数である．MIRA は式（2）の制約に従い，N-best の仮説よりも正解の発音が選ばれるように重みを大きく動かすため，学習データを過学習してしまう．特に発音が誤ったデータ（ノイズデータ）が学習データに多く含まれている場合，誤った方向に重みベクトルを大きく動かしてしまい性能の劣化が予期される．

この過学習の問題を解決するために，我々は構造化 AROW を提案した．この手法では逐次的に仮説 $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$ に対して，以下の関数を最小化する $\boldsymbol{w}_t$ を求める手法である．

$$\begin{aligned} L(\boldsymbol{w}_t, \Sigma_t) = \ & \mathbf{D}_{\mathbf{KL}}(\mathcal{N}(\boldsymbol{w}_t, \Sigma_t)\|\mathcal{N}(\boldsymbol{w}_{t-1}, \Sigma_{t-1})) \\ & + \tfrac{1}{2r}\ell(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{w}_t) + \tfrac{1}{2r}\boldsymbol{o}_t^{\mathrm{T}}\Sigma_t\boldsymbol{o}_t \end{aligned} \tag{3}$$

$\mathbf{D}_{\mathbf{KL}}(\mathcal{N}(\boldsymbol{w}_t, \Sigma_t)\|\mathcal{N}(\boldsymbol{w}_{t-1}, \Sigma_{t-1}))$ は $\boldsymbol{w}_t$ と $\boldsymbol{w}_{t-1}$ の単一多次元ガウス分布間の Kullback-Leibler（KL）ダイバージェンスを意味し，$r > 0$ は汎化能力を調整するハ

* Improvement of Hyperparameter in Adaptive Regularization of Weight Vectors for Grapheme-to-Phoneme Conversion. by Kubo Keigo, Sakriani Sakti, Graham Neubig, Tomoki Toda, Satoshi Nakamura (Nara Institute of Science and Technology)

イパーパラメータ，$\boldsymbol{o}_t$ は正解と仮説の特徴量ベクトルの差ベクトル $\Phi(\boldsymbol{x}_i, \boldsymbol{y}_i) - \Phi(\boldsymbol{x}_i, \hat{\boldsymbol{y}}_n)$ を意味する．共分散行列 Σ の逆行列は各重みの信頼度を表す二次統計量である．$\boldsymbol{w}_t$ は，できるだけ分布を変化させずに，その個々の重みの学習率を考慮して，正解のスコアを増加させ，他の仮説のスコアを減少させるよう更新される．この時，正解のスコアが仮説のスコアよりも高くなることは保証されないが，学習を繰り返すことにより，徐々に多くのデータにおいて正解のスコアが仮説のスコアを上回るようになる．Σ は更新された特徴量（$\boldsymbol{o}_t$ において 0 以外の値を持つ特徴量）の重みの信頼度を増加するように更新される．

Σ を導入し，MIRA では制約であった損失関数を正則化項としてコスト関数に置くことにより，構造化 AROW は過去に何度も更新された（信頼度の高い）識別に重要な重みを極端に動かすことを防ぎ，過学習問題を改善する．特にノイズを含む学習データにおいて性能の劣化を防ぐことが過去の研究により示されている [4].

3 NAROW

AROW と構造化 AROW は重みベクトルの各更新において r に固定の値を設定するため，信頼度が線形に増加して高くなり過ぎる傾向にある．一方で，二値分類の NAROW では r の設定を自身の誤り限界を最小化するように導出する．導出された設定では信頼度が対数的に増加し，かつ現在の学習データに出現する特徴量が十分な信頼度を持つ場合は信頼度を更新しないため，信頼度が高くなり過ぎることを防ぐことができる．これにより，NAROW は重みが早期の段階で不適切な値に収束することを防ぐ．

NAROW の学習では自身の誤り限界を導出するために Follow the Regularized Leader (FTRL) という以下の枠組みに従って重みベクトルを得る．

$$\boldsymbol{w}_t = \arg\min_{\boldsymbol{w}_t} \sum_{i=1}^{t-1} \eta_i \boldsymbol{z}_i^{\mathrm{T}} \boldsymbol{w}_t + f_t(\boldsymbol{w}_t) \tag{4}$$

ここで η_i と $\boldsymbol{z}_i$ は i 回目の更新の学習率と損失関数の劣微分 $\partial \ell_i(\boldsymbol{w}_i)$ である．$f_t(\boldsymbol{w}_t)$ は学習を汎化させるための正則化項（別名，ポテンシャル関数）である．NAROW は各更新 t において $\eta_t = 1$ とし，$f_t(\boldsymbol{w})$ を $\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\Sigma_t^{-1}\boldsymbol{w}$ と定義する．ここで Σ_t^{-1} は各重みの信頼度を表す二次統計量で，$\Sigma_t^{-1} = \Sigma_{t-1}^{-1} + \frac{\Phi(x_t)\Phi(x_t)^{\mathrm{T}}}{r_t}$ に従い更新される．また，$r_t > 0$，$\Sigma_0^{-1} = \boldsymbol{I}$ である．Σ_t^{-1} の更新式は，その r_t が各更新において固定の値 r を持つ場合，AROW の共分散行列 Σ_t の更新式を逆数にした式と同じになる．NAROW における r_t の設定は $bv_t > 1$ の時 $r_t = \frac{v_t}{bv_t - 1}$，その他は $r_t = +\infty$ である．ここで $v_t = \Phi(x_t)^{\mathrm{T}}\Sigma_{t-1}\Phi(x_t) > 0$ は全特徴量の分散具合を示す変数で，$b > 0$ は新しいハイパーパラメータである．この設定は信頼度が高くなり過ぎることを防ぐ．

4 構造化 NAROW

構造化 NAROW もまた自身の誤り限界を最小化するように r_t の設定を導出する．その導出された設定は NAROW と同じ特性を持っており，構造化 AROW の問題を改善することができる．5 節において構造化 NAROW の誤り限界と r_t の設定の導出について説明する．この節では NAROW との違いと構造化 NAROW の具体的な学習手続きを説明する．

Algorithm 1 Follow the Regularized Leader に基づく構造化されたオンライン識別学習（提案手法）

Input: Training dataset $\boldsymbol{D} = \{(\bar{\boldsymbol{x}}_1, \bar{\boldsymbol{y}}_1), ..., (\bar{\boldsymbol{x}}_{|\boldsymbol{D}|}, \bar{\boldsymbol{y}}_{|\boldsymbol{D}|})\}$ and a series of regularizers $f_0, \ldots, f_{T-1}$
Output:weight vector $\boldsymbol{w}_T$
$t = 1, \boldsymbol{\theta}_0 = \boldsymbol{0}$
repeat
  for $i = 1$ **to** $|\boldsymbol{D}|$ **do**
    $\boldsymbol{w}_t = \nabla f_{t-1}^*(\boldsymbol{\theta}_{t-1}) = \Sigma_{t-1}\boldsymbol{\theta}_{t-1}$
    Predict N-best hypotheses $\tilde{\boldsymbol{y}}_1, ..., \tilde{\boldsymbol{y}}_N$ by $\boldsymbol{w}_t^{\mathrm{T}}\Phi(\bar{\boldsymbol{x}}_i, \tilde{\boldsymbol{y}})$
    for $n = 1$ **to** N **do**
      Consider $\boldsymbol{x}_t := \bar{\boldsymbol{x}}_i$, $\boldsymbol{y}_t := \bar{\boldsymbol{y}}_i$, $\hat{\boldsymbol{y}}_t := \tilde{\boldsymbol{y}}_n$ and $\ell_t(\boldsymbol{w}_t) := \max(0, v_t d_t - \boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{o}_t)$
      if $\ell_t(\boldsymbol{w}_t) > 0$ **then**
        $\boldsymbol{z}_t = -\boldsymbol{o}_t \in \partial \ell_t(\boldsymbol{w}_t)$
        $\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \boldsymbol{z}_t$
        $t = t + 1$
        $\boldsymbol{w}_t = \nabla f_{t-1}^*(\boldsymbol{\theta}_{t-1}) = \Sigma_{t-1}\boldsymbol{\theta}_{t-1}$
      end if
    end for
  end for
until Stop condition is met

構造化 NAROW は損失関数 ℓ と Σ_t^{-1} の更新，r_t の計算において，特徴量ベクトル $\Phi(x_t)$ の代わりに正解と仮説の特徴量の差ベクトル $\boldsymbol{o}_t$ を用いる．これは，二値分類の学習に用いられる正解判定項 $y_t(\boldsymbol{w}_t \cdot \Phi(x_t))$（ここで y_t は 1 か -1 を持つ正解ラベルである）を構造学習に拡張すると $\boldsymbol{w}_t \cdot \boldsymbol{o}_t$ と定義できるからである．これらはどちらも，分類が正しい時は正，誤りの時は負の値を取る．また，二値分類の y_t による分類の正誤を判定する役割は $\boldsymbol{o}_t$ 内に含まれている．ただし，$\boldsymbol{w}_t \cdot \boldsymbol{o}_t$ は一つの仮説に対してしか分類の正誤を保証しない．一方で，構造学習では無数の仮説があるため，全仮説に関する $\boldsymbol{o}_t$ を学習に使うことは難しい．そのため，構造化 NAROW では N-best の仮説に関する $\boldsymbol{o}_t$ だけを使用する．

また，構造化 NAROW では $\boldsymbol{w}_t$ を求める際に，正則化項 $f_t(\boldsymbol{w}_t)$ の代わりに，$f_{t-1}(\boldsymbol{w}_t)$ を用いる．その理由は，NAROW ではクラスに依存しない現在のデータの特徴量ベクトル $\Phi(x_t)$ を $f_t(\boldsymbol{w}_t)$ 内の Σ_t^{-1} の更新に用いるため，容易に Σ_t^{-1} を求められるが，構造学習では現在のデータの正解と仮説のクラスに依存する $\boldsymbol{o}_t$ が必要となるため，Σ_t^{-1} を容易に求められないからである．なぜならば，その仮説のクラスは，求めたい $\boldsymbol{w}_t$ により選別されるからである．

構造学習用に拡張された損失関数 ℓ は以下のように定義される．

$$\ell_t(\boldsymbol{w}_t) = \max(0, v_t d_t - \boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{o}_t) \tag{5}$$

ここで，$d_t = d(\boldsymbol{y}_t, \hat{\boldsymbol{y}}_t)$ は正解 $\boldsymbol{y}_t$ を仮説 $\hat{\boldsymbol{y}}_t$ と推定した場合に起こる損失値を表す．本論文では d_t を音素誤り数として定義する．損失関数 ℓ 内に定義されている d_t と v_t の積により，更新対象の全特徴量の分散が高い（まだ更新回数が少ない）場合は損失値が大きくなる．また，損失関数 ℓ の劣微分 $\boldsymbol{z}_t$ は，$\ell_t(\boldsymbol{w}_t) > 0$ の時 $-\boldsymbol{o}_t$，それ以外は $\boldsymbol{0}$ とする．

式（4）を上記の説明を考慮して構造学習に拡張後，導出した重みの更新式は以下の通りである．

$$\boldsymbol{w}_t = \nabla f_{t-1}^{-1}(\boldsymbol{\theta}_{t-1}) = \nabla f_{t-1}^{*}(\boldsymbol{\theta}_{t-1}) = \Sigma_{t-1}\boldsymbol{\theta}_{t-1} \quad (6)$$

ここで $\boldsymbol{\theta}_{t-1} = -\sum_{i=1}^{t-1} \boldsymbol{z}_i$，$\boldsymbol{\theta}_0 = \boldsymbol{0}$，$\nabla f_{t-1}^{*}(\boldsymbol{\theta}_{t-1})$ は，f_{t-1} に対してフェンシェルの共役変換を行った関数の勾配である．f_{t-1}^{*} は $f_{t-1}^{*}(\boldsymbol{\theta}_{t-1}) = \sup_{\boldsymbol{v}} \left\{\boldsymbol{\theta}_{t-1}^{\mathrm{T}}\boldsymbol{v} - f_{t-1}(\boldsymbol{v})\right\} = \frac{1}{2}\boldsymbol{\theta}_{t-1}^{\mathrm{T}}\Sigma_{t-1}\boldsymbol{\theta}_{t-1}$ として定義される（sup は上限を意味する）．N-best 学習に対応した FTRL に基づく構造化 NAROW の学習アルゴリズムを **Algorithm 1** に示す．

5 構造化 NAROW の誤り限界

構造化 NAROW に関する誤り限界を最小化する r_t の設定を導出するために，本節ではオンライン凸最適化に基づく構造化 NAROW の誤り限界を導出する．

5.1 オンライン凸最適化

オンライン凸最適化はポテンシャル関数（正則化項）を通してオンラインアルゴリズムを解析・設計するための方法である．Orabona らは更新 t ごとにポテンシャル関数が変化する FTRL に基づくオンライン凸最適化を用いて NAROW の誤り限界を導出した [7]．構造化 NAROW もその枠組みに従い誤り限界を導出する．

ここで凸解析に関するいくつかの定義を導入する．ノルム $\|\cdot\|$ に関して β 強凸とは $f(\boldsymbol{v}) \geq f(\boldsymbol{u}) + \nabla f(\boldsymbol{u})^{\mathrm{T}}(\boldsymbol{v}-\boldsymbol{u}) + \frac{1}{2}\beta\|\boldsymbol{v}-\boldsymbol{u}\|^2$ を満たす関数のことである．ここで $\boldsymbol{u}, \boldsymbol{v} \in \mathrm{ri}(\mathrm{dom}(f))$ である（$\mathrm{ri}(\mathrm{dom}(f))$ は f の実効定義域の相対的内点を意味する）．3 節で定義された関数 $f_t(\boldsymbol{o}) = \frac{1}{2}\boldsymbol{o}^{\mathrm{T}}\Sigma_t^{-1}\boldsymbol{o}$ はノルム $\|\boldsymbol{o}\|_{f_t}^2 = \boldsymbol{o}^{\mathrm{T}}\Sigma_t^{-1}\boldsymbol{o}$ に関して 1-強凸である．ノルム $\|\cdot\|$ に関する双対ノルム $\|\cdot\|_*$ は $\|\boldsymbol{u}\|_* := \sup\left\{\boldsymbol{u}^{\mathrm{T}}\boldsymbol{v} : \|\boldsymbol{v}\| \leq 1\right\}$ として定義されるノルムである．双対ノルム $\|\cdot\|_*$ に関して β-強凹は $f_*(\boldsymbol{u}+\boldsymbol{v}) \leq f_*(\boldsymbol{u}) + \nabla f_*(\boldsymbol{u})^{\mathrm{T}}\boldsymbol{v} + \frac{1}{2}\beta\|\boldsymbol{v}\|_*^2$ を満たす関数である．4 節において定義された $f_t^*(\boldsymbol{o}) = \frac{1}{2}\boldsymbol{o}^{\mathrm{T}}\Sigma_t\boldsymbol{o}$ はノルム $\|\boldsymbol{o}\|_{f_t^*}^2 = \boldsymbol{o}^{\mathrm{T}}\Sigma_t\boldsymbol{o}$ に関して 1-強凹である．β-強凸・凹は誤り限界を導出するために重要な特性である．

5.2 誤り限界の導出

この節では誤り限界の導出に関して簡単に説明する．導出の詳細は，2 値分類と構造学習の違いはあるが（正則化項における Σ_{t-1}^{-1} と Σ_t^{-1} の違い），[7] が参考になる．最初に以下の条件を導入する．

$$d_t - \ell_t(\boldsymbol{u}) \leq -\boldsymbol{u}^{\mathrm{T}}\boldsymbol{z}_t; \forall \boldsymbol{u} \in S, v_t \geq 1 \quad (7)$$

ここで $\boldsymbol{z}_t$ は $\ell_t(\boldsymbol{w}_t) > 0$ を満たす劣微分の値である．また，ここで $v_t \geq 1$ が全ての t で必ず満たされると仮定する．その例外（$v_t < 1$）に関しては 5.3 節で説明する．[7] における補題 1，3 節と 4 節における設定，式（7），$f_t(\lambda\boldsymbol{u}) \leq \lambda^2 f_t(\boldsymbol{u})$ から，以下の誤り限界に関する不等式を得ることができる．

$$\begin{aligned}\sum_{t\in M\cup U}(d_t - \ell_t(\boldsymbol{u})) &= D + \sum_{t\in U} d_t - \sum_{t\in M\cup U}\ell_t(\boldsymbol{u}) \\ &\leq \frac{\lambda\|\boldsymbol{u}\|^2}{2} + \sum_{t\in M\cup U}\Big(\frac{\lambda(\boldsymbol{u}^{\mathrm{T}}\boldsymbol{x}_t)^2}{2r_t} \\ &\quad + \frac{v_t r_t}{2\lambda(r_t+v_t)} - \frac{m_t^2}{2\lambda(r_t+v_t)} + \frac{m_t}{\lambda}\Big), \end{aligned} \quad (8)$$

ここで $m_t = \boldsymbol{o}_t^{\mathrm{T}}\Sigma_{t-1}\boldsymbol{\theta}_{t-1}$，$\boldsymbol{u}$ は任意の重みベクトル，$\lambda \geq 0$ は任意のスケール因子である．D は分類誤りの数，M は推定を誤ったデータ数，U は正しい推定を行ったが正解のスコアが仮説のスコアよりも $v_t d_t$ だけ高くなかったデータ数を表す．

5.3 ハイパーパラメータ r_t の選択

我々は上記の誤り限界の右辺を最小化する（分類誤りを最小化する）ハイパーパラメータ r_t 選択したい．Orabona らは 2 値分類において，式（8）における $\frac{\lambda(\boldsymbol{u}^{\mathrm{T}}\boldsymbol{x}_t)^2}{2r_t} + \frac{v_t r_t}{2\lambda(r_t+v_t)}$ の部分を最小化することを焦点に当てている [7]．v_t が十分に小さい時，その第 2 項は 0 に近くなり無視することができるため，第 1 項を最小化するために $r_t = +\infty$ とする．v_t が大きい場合は第 2 項を無視できないため，それに合わせて r_t が小さくなるよう $r_t = \frac{v_t}{bv_t - 1}$ とする．v_t が十分に小さいかどうかは $bv_t > 1$ を満たさないかどうかで決定する．ハイパーパラメータ $b > 0$ は第 1 項と第 2 項の最小化に関するトレードオフを制御する．この設定は信頼度を対数的に増加させ，更新される全特徴量の信頼度が十分に高い場合は信頼度を更新しないということを意味しており，これにより信頼度が高くなりすぎることを防ぐ．また，この設定は誤り限界を最小化するように導出されるため，解析的にも，固定の値を設定するよりも誤りが少ない分類を行うことが分かる．

この設定を構造学習にも採用して式（8）の r_t に代入後，λ で偏微分し，それを 0 と置いて，最適な λ を得ることにより，構造化 NAROW に関する誤り限界が以下のように得られる．

$$\begin{aligned} D \leq \sum_{t\in M\cup U}\ell_t(\boldsymbol{u}) + \sqrt{\|\boldsymbol{u}\|^2 + \sum_{t:bv_t>1}\frac{bv_t(\boldsymbol{u}^{\mathrm{T}}\boldsymbol{x}_t)^2}{(r_t+v_t)}} \\ \times\sqrt{\sum_{t\in M\cup U}(\min(\frac{1}{b}, v_t) - \frac{m_t^2}{(r_t+v_t)} + 2m_t)} - \sum_{t\in U} d_t \end{aligned} \quad (9)$$

g2p 変換の特徴量数は膨大なため，実際は全共分散行列 Σ_t^{-1} の代わりに対角行列 $\mathrm{diag}\left\{\Sigma_t^{-1}\right\}$ を用いることに注意する．式（7）における $v_t \geq 1$ に関して，b を小さく設定すれば v_t は小さくなり過ぎることはない．それゆえ，b を小さく設定することによりその不等式を満たすようにする．[1]

6 評価実験

g2p 変換タスクを用いた実験により構造化 NAROW を評価した．表 1 はこの実験において用いたデータセットのデータ名（Dataset），出現する書記素と音素の種類数（g/p: g が書記素，p が音素の種類数に対応），学習データ数（Train），開発データ数（Dev），テストデータ数（Test），交差検定の回数（K-fold）を示している．表 1 におけるデータセットに関して，NETtalk，Brulex，Beep は，Pascal Letter-to-Phoneme Conversion Challenge[2] から得た単語の発音辞書である．また，CMUdict[3]，Celex[4]もまた単語の発音辞書である．文献 [1] の実験で用いられているデータセット（NETtalk，Brulex，Beep，CMUdict）において，我々は，学習データから開発データをランダムに選んだことを除いて，書記素列が 1 文字で構成されるといった例外データの取り除き方，学習データ数（＋開発データ数）とテストデータ数の割合に関して，文献 [1] の実験の再現を試みた．

[1] g2p 変換は豊富な特徴量を用いるため v_t の値が高くなり，その不等式はほぼ確実に満たされることに注意．

[2] http://pascallin.ecs.soton.ac.uk/Challenges/PRONALSYL/Datasets

[3] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[4] http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC96L14

Table 1　g2p 変換タスクの評価実験で使用するデータセット.

Dataset	g/p	Vocabulary size			
		Train	Dev	Test	K-fold
NETtalk (English)	26/50	17595	1000	1000	10
Brulex (French)	40/39	23353	1373	2747	5
CELEX (English)	26/53	39995	15000	5000	1
CMUdict (English)	27/39	100886	5941	12000	2
Beep (English)	26/44	169823	8938	19862	1

Table 2　各手法において設定が必要な特徴量とパラメータ.

	JOINT	MIRA	SAROW	SNAROW
joint n-gram	7	5	5	5
context window	-	6	6	6
N-best hypotheses	-	5	5	5
hyperparameter r	-	-	500, 1000,1500	-
hyperparameter b	-	-	-	0.0075, 0.01,0.0125
beam width	-	50	50	50

評価手法は Sequitur[5] に実装されている書記素列と音素列の結合 N-gram の生成モデルである結合系列モデル（JOINT）と DirecTL+[6]に実装されている g2p 変換のための MIRA に基づくオンライン識別学習（MIRA），構造化 AROW（SAROW），提案手法である構造化 NAROW（SNAROW）を用いた．表 2 はそれらの特徴量や設定が必要なパラメータの詳細を示している．いくつかのパラメータの設定は過去の研究 [4] に基づいている．学習回数とハイパーパラメータ r と b は開発データの音素誤り率を最小にする値が用いられる．SNAROW と MIRA，SAROW の前処理として必要とされる書記素列と音素列の最小単位を決めるアライメントには mpaligner[7] に実装されている制約なし多対多アライメント手法を用いた [8].

表 3 に g2p 変換タスクにおける評価結果を示す．PER は音素誤り率，WER は単語誤り率を意味する．NETtalk，Brulex，CMUdict の結果は各交差検定における結果の平均である．有意差検定には Paired Bootstrap Resampling [9] を使用し，有意水準 0.05 で検定した．太字はその評価指標において最も性能が高かった手法とその手法に対して有意差がなかった手法である．表 3 から，PER に関して，提案手法は CMUdict と Beep 以外の全てのデータセットにおいて，有意な差で他の手法を改善している．MIRA と SAROW に対する誤り削減率は 0.7-9.2%であった．このことから，提案手法は g2p 変換タスクにおいて，今回比較した手法の中で最も有効な手法であることが分かった.

7　まとめ

本論文では 2 値分類の NAROW を構造学習に拡張した構造化 NAROW を提案し，g2p 変換タスクにおいてそれを評価した．構造化 NAROW はハイパーパラメータ r_t を信頼度を高くし過ぎないように設定することで，重みが早期の段階で不適切な値に収束してしまう構造化 AROW の問題を解決した．評価実験において，我々の提案手法は様々な辞書の音素誤り率において 0.7-9.2%の誤り削減率を得て，有意に MIRA と構造化 AROW を改善した.

Table 3　g2p 変換タスクにおける評価実験の結果.

Dataset	Measure	JOINT	MIRA	SAROW	SNAROW
NETtalk	PER(%)	7.71	6.70	6.75	**6.53**
	WER(%)	31.6	**28.18**	28.66	**27.97**
Brulex	PER(%)	1.26	1.03	1.09	**0.99**
	WER(%)	6.57	**5.24**	5.59	**5.14**
CELEX English	PER(%)	2.62	2.39	2.51	**2.30**
	WER(%)	12.15	**11.07**	11.81	**11.17**
CMUdict	PER(%)	6.77	6.19	**6.15**	**6.11**
	WER(%)	28.55	**26.35**	**26.48**	**26.46**
Beep	PER(%)	2.26	2.35	**2.19**	**2.16**
	WER(%)	12.24	12.60	**11.73**	**11.57**

謝辞　本研究の一部は，JSPS 科研費 24240032 および（独）情報通信研究機構の委託研究「知識・言語グリッドに基づくアジア医療交流支援システムの研究開発」の助成を受けたものである。

参考文献

[1] M. Bisani and H. Ney, “Joint-sequence models for grapheme-to-phoneme conversion,” Speech Communication, vol.50, no.5, pp.434–451, 2008.

[2] S. Jiampojamarn and G. Kondrak, “Online discriminative training for grapheme-to-phoneme conversion,” Proc. INTERSPEECH, pp.1303–1306, 2009.

[3] S. Jiampojamarn, C. Cherry, and G. Kondrak, “Integrating joint n-gram features into a discriminative training framework,” Proc. NAACL-HLT, pp.697–700, 2010.

[4] K. Kubo, S. Sakti, G. Neubig, T. Toda, and S. Nakamura, “Grapheme-to-phoneme conversion based on adaptive regularization of weight vectors,” Proc. INTERSPEECH, pp.1946–1950, 2013.

[5] K. Crammer and Y. Singer, “Ultraconservative online algorithms for multiclass problems,” Journal of Machine Learning Research, vol.3, pp.951–991, 2003.

[6] K. Crammer, A. Kulesza, and M. Dredze, “Adaptive regularization of weight vectors,” Advances In Neural Information Processing Systems, vol.23, pp.414–422, 2009.

[7] F. Orabona and K. Crammer, “New adaptive algorithms for online classification,” Proc. NIPS, pp.1840–1848, 2010.

[8] K. Kubo, H. Kawanami, H. Saruwatari, and K. Shikano, “Unconstrained many-to-many alignment for automatic pronunciation annotation,” Proc. APSIPA, pp.1–4, 2011.

[9] P. Koehn, “Statistical significance tests for machine translation evaluation.,” EMNLP, pp.388–395, 2004.

[5] http://sequitur.info/
[6] http://code.google.com/p/directl-p/
[7] http://sourceforge.jp/projects/mpaligner/