PAPER Special Section on Advances in Modeling for Real-world Speech Information Processing and its Application

Structured Adaptive Regularization of Weight Vectors for a Robust Grapheme-to-Phoneme Conversion Model

Keigo KUBO[†a)], ***Student Member***, **Sakriani SAKTI**[†], ***Member***, **Graham NEUBIG**[†], ***Nonmember***, **Tomoki TODA**[†], ***and*** **Satoshi NAKAMURA**[†], ***Members***

SUMMARY Grapheme-to-phoneme (g2p) conversion, used to estimate the pronunciations of out-of-vocabulary (OOV) words, is a highly important part of recognition systems, as well as text-to-speech systems. The current state-of-the-art approach in g2p conversion is structured learning based on the Margin Infused Relaxed Algorithm (MIRA), which is an online discriminative training method for multiclass classification. However, it is known that the aggressive weight update method of MIRA is prone to overfitting, even if the current example is an outlier or noisy. Adaptive Regularization of Weight Vectors (AROW) has been proposed to resolve this problem for binary classification. In addition, AROW's update rule is simpler and more efficient than that of MIRA, allowing for more efficient training. Although AROW has these advantages, it has not been applied to g2p conversion yet. In this paper, we first apply AROW on g2p conversion task which is structured learning problem. In an evaluation that employed a dataset generated from the collective knowledge on the Web, our proposed approach achieves a 6.8% error reduction rate compared to MIRA in terms of phoneme error rate. Also the learning time of our proposed approach was shorter than that of MIRA in almost datasets.

key words: *g2p conversion, out-of-vocabulary word, online discriminative training, structured learning, AROW*

1. Introduction

Advances in speech recognition technology have made it possible to attempt large-scale, open-domain, data-driven approaches. Out-of-vocabulary (OOV) words are the bottleneck in such speech recognition systems, and the need for robust pronunciation annotation has been increasing. For example, voice search applications have attracted attention because of an increased demand for mobile device interfaces. A variety of words, such as proper nouns and brand-new words, must be dealt with in these applications. It is important to update the language model and pronunciation dictionary to accommodate these terms. We can collect sentences from Web text resources to train the LM, but the pronunciation of some of those words may be unknown. Therefore, grapheme-to-phoneme (g2p) conversion must be used to estimate the pronunciations of out-of-vocabulary (OOV) words in speech recognition systems [1] or text-to-speech systems [2].

The training procedure for g2p conversion has an alignment step and a parameter estimation step. The alignment step performs segmentation and maps between graphemes and phonemes on the training data. One-to-one [3] and many-to-many [4]–[7] alignment methods have been proposed. The parameter estimation step learns the parameters used for g2p conversion from training data segmented and mapped by the alignment step. Rule-based approaches [8] and statistical approaches based on methods such as neural networks [9], decision trees [10], and maximum entropy [11] have been proposed.

There are two major statistical approaches for parameter estimation of g2p conversion: the joint sequence model [12], [13] and structured learning based on the Margin Infused Relaxed Algorithm (MIRA) [14]. The joint sequence model is a generative model employing joint n-grams for graphemes and phonemes. This approch performs the alignment step and the parameter estimation step at the same time. MIRA is an online discriminative training method for models of multiclass classification that learns parameters that correctly classify the current instance with a sufficient margin. MIRA has also been expanded to structured learning problems for which there are an extremely large number of candidate answers, such as g2p conversion, and employed to the parameter estimation step of g2p conversion [15], [16]. Previous reports on MIRA-based g2p note that it outperforms the joint sequence model in terms of word error rate on g2p tasks. However, MIRA is also prone to overfitting, as it updates parameters to correctly classify the current example, even if the current example is an outlier or noisy.

Recently, methods have been proposed that employ pronunciations from the collective knowledge on the World Wide Web as training data for g2p models without a cross-check of language experts [17]. In this case, the training data is expected to include a lot of noisy data, and empirically, in [17], this degrades the performance of the speech recognition system in exchange for improvements of cost and time required for dictionary construction. When this sort of noisy data is used to train a g2p system, it is extremely important to have an approach that is highly accurate and robust to overfitting.

Adaptive Regularization of Weight Vectors (AROW) [18] is another online discriminative training method for binary classification that has been proposed as an approach to resolve overfitting. This is achieved by gradually learning parameters to correctly classify the training data, without guaranteeing that the current example is correctly classified. In addition, AROW's update rule is simpler than that of

Manuscript received September 27, 2013.

Manuscript revised January 18, 2014.

[†]The authors are with the Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma-shi, 630–0192 Japan.

a) E-mail: keigo-k@is.naist.jp

DOI: 10.1587/transinf.E97.D.1468

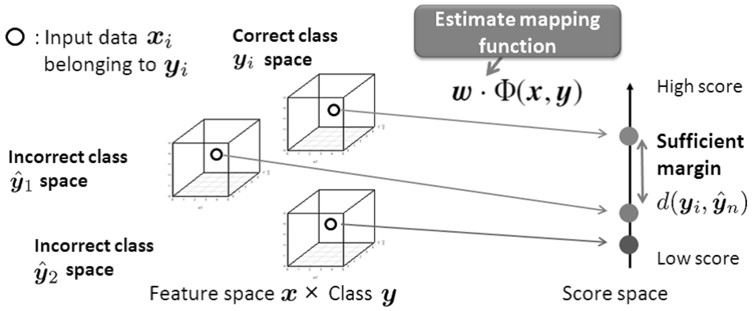
○ : Input data $\boldsymbol{x}_i$ belonging to $\boldsymbol{y}_i$
Correct class $\boldsymbol{y}_i$ space
Estimate mapping function
$\boldsymbol{w} \cdot \Phi(\boldsymbol{x}, \boldsymbol{y})$
High score
Incorrect class $\hat{\boldsymbol{y}}_1$ space
Sufficient margin
$d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n)$
Incorrect class $\hat{\boldsymbol{y}}_2$ space
Low score
Feature space $\boldsymbol{x}$ × Class $\boldsymbol{y}$
Score space

Fig. 1 Image of a maximum margin method for multiclass classification. In the context of g2p conversion, $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ denote a word and the correct pronunciation in i-th data respectively. Also $\hat{\boldsymbol{y}}_1$ and $\hat{\boldsymbol{y}}_2$ denote inferable other pronunciations (wrong pronunciations) in the word $\boldsymbol{x}_i$.

MIRA, allowing for more efficient training. In multiple binary classification tasks, AROW has been shown to outperform the Passive-Aggressive (PA) algorithm [19] which can be regarded as the binary classification equivalent of MIRA.

In this paper, we first apply AROW to the g2p conversion task which is a structured learning problem. We evaluate the proposed approach on various g2p tasks including collective knowledge data such as wiktionary which is a collaboratively constructed dictionary on the World Wide Web, comparing with the joint sequence model and structured learning based on MIRA. Note that this is an extension of our previous work [20], with a fuller and more clear exposition, and greatly expanded experiments including real noisy data from the web, and data from multiple languages.

The rest of this paper is organized as follows. In Sect. 2, we describe g2p conversion based on linear classifiers, which are employed in the existing method based on MIRA and our proposed approach based on AROW. The existing structured learning approach based on MIRA is described in Sect. 3. In Sect. 4, we describe AROW and Confidence Weighted Algorithm (CW) [21], [22] which is a predecessor approach of AROW, as binary classification methods. Structured AROW which is our proposed approach and extends AROW to structured learning for g2p conversion is described in Sect. 5. In Sect. 6, we report on an evaluation experiment for our proposed approach on various g2p tasks. Finally, Sect. 7 states our conclusion.

2. G2p Conversion Based on Linear Classifiers

We define g2p conversion as a process of converting a grapheme sequence $\boldsymbol{x}$ into a phoneme sequence $\boldsymbol{y}$. To obtain a correct phoneme sequence $\boldsymbol{y}$ from a grapheme sequence $\boldsymbol{x}$, we employ a linear classifier defined as

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

where $\boldsymbol{w}$ indicates the classifier's weight vector and $\Phi(\boldsymbol{x}, \boldsymbol{y})$ indicates a feature vector which consists of arbitrary values such as frequencies of joint n-gram features [16] on $\boldsymbol{x}$ and $\boldsymbol{y}$. In Eq. (1), $\hat{\boldsymbol{y}}$ can be efficiently obtained using dynamic programming. Structured learning can be employed to obtain a $\boldsymbol{w}$ that allows for accurate prediction of the correct phoneme sequence in this framework.

3. Structured Learning Using MIRA

MIRA is a kind of a maximum margin method for multiclass classification as shown in Fig. 1. Given data $\boldsymbol{x}_i$ belonging to class $\boldsymbol{y}_i$, MIRA assigns a vector in feature space to each inferable class including both correct classes and incorrect classes, where the correct class corresponds to the label in the manually annotated data, and the incorrect class means the rest of the inferable labels for the data. In the context of g2p conversion, these are the correct pronunciation and wrong pronunciations respectively. MIRA maps the feature vector in each inferable class to score space by a mapping function. Then, MIRA estimates parameters in the mapping function so that the correct class scores higher than the incorrect class with a sufficient margin.

Structured learning based on MIRA for g2p has been proposed in [15]. When the i-th example $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and n-best hypotheses $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$ produced by $\boldsymbol{w}_{t-1} \cdot \Phi(\boldsymbol{x}_i, \hat{\boldsymbol{y}})$ are given, it updates the current weight vector $\boldsymbol{w}_{t-1}$ by solving the constrained optimization problem defined as

$$\begin{aligned} &\min_{\Delta\boldsymbol{w}} \frac{1}{2}\|\Delta\boldsymbol{w}\|^2 \\ &\quad \text{s.t.} \quad \forall n \\ &\qquad (\boldsymbol{w}_{t-1} + \Delta\boldsymbol{w}) \cdot \boldsymbol{u}_{in} \geq d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n) \end{aligned} \tag{2}$$

where $\Delta\boldsymbol{w}$ indicates the update vector for weights. The updated weight vector $\boldsymbol{w}_t$ defined as

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \Delta\boldsymbol{w} \tag{3}$$

and $\boldsymbol{u}_{in}$ is defined as $\Phi(\boldsymbol{x}_i, \boldsymbol{y}_i) - \Phi(\boldsymbol{x}_i, \hat{\boldsymbol{y}}_n)$ which is the difference vector between the feature vector of a correct target sequence $\boldsymbol{y}_i$ and a feature vector of the hypothesis $\hat{\boldsymbol{y}}_n$. The $d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n)$ indicates the loss incurred by incorrectly classifying $\boldsymbol{y}_i$ as $\hat{\boldsymbol{y}}_n$. In g2p conversion, the source sequence $\boldsymbol{x}_i$ and the target sequence $\boldsymbol{y}_i$ are a grapheme sequence and a phoneme sequence respectively, and the phoneme error rate of prediction is used as the loss $d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n)$. As in Eq. (2), structured learning based on MIRA employs n-best hypotheses $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$ in training and finds the updated weight vector $\boldsymbol{w}_t$ that correctly classifies the current example $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ with a sufficient margin proportional to the loss of each hypothesis $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$.

Algorithm 1 Structured learning based on MIRA

```
Input:Training dataset $\boldsymbol{D} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), \dots, (\boldsymbol{x}_{|\boldsymbol{D}|}, \boldsymbol{y}_{|\boldsymbol{D}|})\}$
Output:$\boldsymbol{w}$
$\boldsymbol{w} = \boldsymbol{0}$
repeat
    for $i = 1$ to $|\boldsymbol{D}|$ do
        Predict $n$-best hypotheses $\hat{\boldsymbol{y}}_1, \dots, \hat{\boldsymbol{y}}_N$ by $\boldsymbol{w} \cdot \Phi(\boldsymbol{x}_i, \hat{\boldsymbol{y}})$
        Update $\boldsymbol{w}$ by solving the constrained optimization problem of Eq. (2)
    end for
until Stop condition is met
```

When the constrained optimization problem in Eq. (2) is solved using Lagrange multipliers, the dual problem can be obtained as follows:

$$\max_{\alpha_1,\dots,\alpha_N} \frac{1}{2} \sum_n \sum_m \alpha_n \alpha_m \boldsymbol{u}_{in} \cdot \boldsymbol{u}_{im} + \sum_n \alpha_n (d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n) - \boldsymbol{w}_{t-1} \cdot \boldsymbol{u}_{in}) \qquad (4)$$

where $\alpha_1, \dots, \alpha_N$ indicates optimized Lagrange multipliers and $\Delta\boldsymbol{w}$ is expressed with $\alpha_1, \dots, \alpha_N$ defined as

$$\Delta\boldsymbol{w} = \sum_n \alpha_n \boldsymbol{u}_{in}. \qquad (5)$$

Equation (4) is a quadratic programming problem having parameters $\alpha_1, \dots, \alpha_N$. When the parameters are optimized by numerical computation [23], the updated weight vector $\boldsymbol{w}_t$ can be obtained from Eq. (3) and Eq. (5). If there are many parameters to be optimized, the quadratic programming problem is difficult to solve in terms of computation cost. For a method based on MIRA, the number of parameters to be optimized is equal to the number of hypotheses employed in update. Therefore, to decrease the computational cost, MIRA is generally used in the context of online, instead of batch, learning.

The procedure of structured learning based on MIRA is shown in **Algorithm 1**. In [15], n-best hypotheses $\hat{\boldsymbol{y}}_1, \dots, \hat{\boldsymbol{y}}_N$ are approximately predicted by beam-search pruning based on a monotone phrasal decoder [24].

One known weakness of MIRA is that it is prone to overfitting. Even if the current example is an outlier or noisy, MIRA must classify the current example correctly, and will move the weights as much as is necessary to do so. This can degrade system performance by causing overfitting. To resolve this problem, we propose Structured AROW, which is more robust in the face of overfitting compared with MIRA.

4. AROW and CW for Binary Classification

CW and AROW are online discriminative training methods and a kind of a maximum margin method for binary classification. Both methods assume that the weight vector $\boldsymbol{w}$ follows a multi-dimensional Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$, where d is the number of features in the model. During prediction, CW and AROW employ the expectation of the weight vector $\mathrm{E}[\boldsymbol{w}] = \boldsymbol{\mu}$ instead of the weight vector $\boldsymbol{w}$. By considering the variance and covariance, CW and AROW individually control the amount of each feature weight updated after each example. Because the current weight of the features that have frequently occurred and been updated in the past has high confidence, they are not moved excessively on any update. In contrast, because the current weight of the features that have rarely been updated in the past does not have high confidence, they are widely moved on update. This property, which MIRA does not have, prevents the important weights that have high confidence from widely moving in directions that degrade the system performance in the presence of outliers. On the other hand, it widely moves rare features in contrast to batch learning with a regularization. In multiple binary classification tasks for natural language processing (NLP), it has been shown that because rare features are often very informative in NLP, online learning algorithms that have that property have been shown to outperform batch learning of maximum entropy classifiers and support vector machines [22]. Also, because online learning generally runs faster and uses less memory than batch learning, we focus on online learning rather than batch learning in this paper. In the rest of this section, we describe differences between CW and AROW.

4.1 CW

When the i-th example $(\boldsymbol{x}_i, y_i)$ is given, CW obtains an updated distribution $\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t)$ for the weight vector by solving the constrained optimization problem defined as

$$\begin{aligned}(\boldsymbol{\mu}_t, \Sigma_t) = \; & \min_{\boldsymbol{\mu}_t, \Sigma_t} \mathbf{D_{KL}}(\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t) || \mathcal{N}(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})) \\ & \text{s.t.} \quad \Pr_{\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t)}[y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i) \geq 0] \geq \eta \end{aligned} \qquad (6)$$

where $\mathcal{N}(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})$ is the current distribution for the weight vector, $\mathbf{D_{KL}}(\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t) || \mathcal{N}(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1}))$ indicates the Kullback-Leibler (**KL**) divergence between the updated distribution and the current distribution, and $\eta \in (0.5, 1]$ is a hyperparameter controlling the margin. Note that $\boldsymbol{x}_i$ and $y_i \in \{-1, +1\}$ here indicate the i-th input vector and the i-th correct label (-1 or 1) respectively, whereas in our description for structured learning based on MIRA and AROW we assume $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ to be the source sequence and the target sequence respectively. As in Eq. (6), CW finds the updated distribution that is closest to the previous distribution while satisfying the constraint that the current example $(\boldsymbol{x}_i, y_i)$ is correctly classified with at least probability $\eta \in (0.5, 1]$. The learning of CW converges quickly, as the constraint of CW forces CW to find the distribution that correctly classifies the current example $(\boldsymbol{x}_i, y_i)$ with at least probability $\eta \in (0.5, 1]$. However, like MIRA, this aggressive learning causes overfitting, since CW has the possibility to widely move even a reliable weight to satisfy this constraint.

4.2 AROW

To avoid this problem of MIRA and CW, AROW recasts terms for the constraint of CW as regularizers. The distribution found by AROW does not guarantee that the current example $(\boldsymbol{x}_i, y_i)$ is correctly classified. However, the training data comes closer to being correctly classified each time the distribution is updated, and even when an outlier appears, AROW does not widely move the reliable weights in a direction that degrades the system performance.

AROW obtains the updated distribution for the weight vector by solving the unconstrained optimization problem defined as

$$(\boldsymbol{\mu}_t, \Sigma_t) = \min_{\boldsymbol{\mu}_t, \Sigma_t} \mathbf{D_{KL}}(\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t) \| \mathcal{N}(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})) + \frac{1}{2r}\ell_{h^2}(\boldsymbol{x}_i, y_i, \boldsymbol{\mu}_t) + \frac{1}{2r}\boldsymbol{x}_i^{\mathrm{T}}\Sigma_t\boldsymbol{x}_i \tag{7}$$

where r is a hyperparameter that has the constraint $r > 0$, and controls the update amount for $\boldsymbol{\mu}$ and Σ. Also, $\ell_{h^2}(\boldsymbol{x}_i, y_i, \boldsymbol{\mu}_t)$ is the loss function defined as

$$\ell_{h^2}(\boldsymbol{x}_i, y_i, \boldsymbol{\mu}_t) = (\max\{0, 1 - y_i(\boldsymbol{\mu}_t \cdot \boldsymbol{x}_i)\})^2. \tag{8}$$

Solving Eq. (7) is equivalent to finding the distribution that decreases the loss function value and variances of each feature that occurred, while avoiding changing the previous distribution as much as possible. In multiple binary classification tasks for NLP, AROW has been shown to outperform CW and PA [18].

5. Structured AROW

We propose Structured AROW to extend AROW to structured learning. Structured AROW is also a kind of a maximum margin method such as shown in Fig. 1. In contrast to MIRA, Structured AROW gradually learns parameters to correctly classify the current data with a sufficient margin.

In order to extend binary classification to structured learning, we consider differences between the two settings. The first difference is how to judge whether a prediction is true or not. In binary classification, because the two classes are represented by positive and negative for $\boldsymbol{w} \cdot \boldsymbol{x}_i$, the prediction is judged to be true when $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i)$ is positive, where y_i indicates the correct class with 1 (positive) or -1 (negative). In structured learning, because it can not represent all classes only with positive and negative, the judgment is relaxed to whether a correct class $\boldsymbol{y}_i$ scores higher than a certain hypothesis $\hat{\boldsymbol{y}}_n$ or not. It is formalized as $\boldsymbol{w} \cdot \boldsymbol{u}_{in} = \boldsymbol{w} \cdot \Phi(\boldsymbol{x}_i, \boldsymbol{y}_i) - \boldsymbol{w} \cdot \Phi(\boldsymbol{x}_i, \hat{\boldsymbol{y}}_n)$. When $\boldsymbol{w} \cdot \boldsymbol{u}_{in}$ is positive, the prediction is judged to be true over a hypothesis $\hat{\boldsymbol{y}}_n$. Note that the relaxed judgment does not guarantee that the prediction selects a correct class correctly from all classes. The second difference is that structured learning handles countless classes. Thus, it is difficult to learn to reduce output values of loss functions of a current correct class over classes in all inferable hypotheses. So, to extend AROW to Structured AROW, we replace $y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i)$ to $\boldsymbol{w} \cdot \boldsymbol{u}_{in}$ and limit our updates to classes in n-best hypotheses.

Given the i-th data $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and the n-best hypotheses $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$, Structured AROW updates an distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ for $\hat{\boldsymbol{y}}_n; n = 1, \ldots, N$ sequentially, to minimize the objective function defined as

$$L(\boldsymbol{\mu}_t, \Sigma_t) = \mathbf{D_{KL}}(\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t) \| \mathcal{N}(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})) + \frac{1}{2r}\ell_{h^2}(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{\mu}_t) + \frac{1}{2r}\boldsymbol{u}_{in}^{\mathrm{T}}\Sigma_t\boldsymbol{u}_{in} \tag{9}$$

where r is a hyperparameter that has the constraint $r > 0$ as before. And $\ell_{h^2}(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{\mu}_t)$ is the loss function defined as

$$\ell_{h^2}(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{\mu}_t) = (\max\{0, d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n) - \boldsymbol{\mu}_t \cdot \boldsymbol{u}_{in}\})^2. \tag{10}$$

By partially differentiating Eq. (9) with $\boldsymbol{\mu}_t$ and setting this derivative to 0 so that we minimize Eq. (9), the update formula for $\boldsymbol{\mu}_t$ of structured learning based on AROW is as follows:

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \frac{\max\{0, d(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_n) - \boldsymbol{\mu}_{t-1} \cdot \boldsymbol{u}_{in}\}}{\boldsymbol{u}_{in}^{\mathrm{T}}\Sigma_{t-1}\boldsymbol{u}_{in} + r}\Sigma_{t-1}\boldsymbol{u}_{in}. \tag{11}$$

As the full covariance matrix can not be handled as the number of features in g2p conversion is enormous, we assume Σ_t to be a diagonal matrix, as is standard for traditional CW or AROW. We partially differentiate the objective function of Eq. (9) with the p-th diagonal element $(\Sigma_t)_{p,p}$ of Σ_t to obtain the update formula for Σ_t, and then we set the equation to be 0 as follows:

$$\frac{\partial}{\partial(\Sigma_t)_{p,p}}L(\boldsymbol{\mu}_t, \Sigma_t) = \frac{1}{2}\left(\frac{1}{(\Sigma_{t-1})_{p,p}} - \frac{1}{(\Sigma_t)_{p,p}} + \frac{(\boldsymbol{u}_{in})_p^2}{r}\right) = 0 \tag{12}$$

where $(\boldsymbol{u}_{in})_p$ indicates the p-th feature value of the $\boldsymbol{u}_{in}$. We arrange the above equation to solve $(\Sigma_t)_{p,p}$ as follows:

$$(\Sigma_t)_{p,p} = \frac{r(\Sigma_{t-1})_{p,p}}{r + (\boldsymbol{u}_{in})_p^2(\Sigma_{t-1})_{p,p}}. \tag{13}$$

Each diagonal element $(\Sigma_t)_{p,p}$ for $p = 1, \ldots, d$ is updated by Eq. (13). Also when $\ell_{h^2}(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{\mu}_{t-1})$ is equal to 0, $\boldsymbol{\mu}_{t-1}$ and Σ_{t-1} are not updated.

The procedure of Structured AROW is shown in **Algorithm 2**. $\boldsymbol{\mu}$ and Σ are initialized with the zero vector and identity matrix respectively. From $(\Sigma_0)_{p,p} = 1$, $r > 0$ and Eq. (13), $(\Sigma_{t-1})_{p,p} \geq (\Sigma_t)_{p,p}$ for all t holds. When $(\Sigma_t)_{p,p} = 0$,

Algorithm 2 Structured AROW

```
Input: Training dataset $\boldsymbol{D} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_{|\boldsymbol{D}|}, \boldsymbol{y}_{|\boldsymbol{D}|})\}$
Output: $\boldsymbol{\mu}$ as weight vector $\boldsymbol{w}$
$\boldsymbol{\mu} = \boldsymbol{0}, \Sigma = \boldsymbol{I}$
repeat
    for $i = 1$ to $|\boldsymbol{D}|$ do
        Predict $n$-best hypotheses $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$ by $\boldsymbol{\mu} \cdot \Phi(\boldsymbol{x}_i, \hat{\boldsymbol{y}})$
        for $n = 1$ to $N$ do
            if $\ell_{h^2}(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{\mu}) > 0$ then
                Update $\boldsymbol{\mu}$ and $\Sigma$ by Eq. (11) and Eq. (13) respectively
            end if
        end for
    end for
until Stop condition is met
```

Table 1 *Dataset used in the preliminary experiment on the g2p task. g/p indicates the number of grapheme and phoneme symbols. Noisy indicates the number of artificial noisy data.*

Dataset	g/p	Vocabulary size			
		Train (Noisy)	Dev	Test	K-fold
NETtalk	26/50	17595 (0)	1000	1000	10
Noisy NETtalk	26/50	17595 (1760)	1000	1000	10

Table 2 *Parameter settings for the preliminary experiment. They were optimized for each method on the development data, with the parameters employed at least once in each cross-validation fold in bold.*

	Joint Sequence	MIRA	Structured AROW
joint n-gram	**5**, **6**, **7**, 8, **9**, 10	Follow Joint Sequence	Follow Joint Sequence
context window	-	**4**, **5**, **6**	Follow MIRA
n-best hypotheses	-	1, 3, **5**	Follow MIRA
hyperpara- meter *r*	-	-	**500**, **1000**, **1500**
beam width	-	**150**	**150**

the p-th feature weight of the $\boldsymbol{\mu}$ is fixed. Therefore, the convergence of **Algorithm 2** is guaranteed. In **Algorithm 2**, n-best hypotheses $\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_N$ are also predicted by beam-search pruning based on a monotone phrasal decoder [24], similarly to [15]. The update process for the $\boldsymbol{\mu}$ and the Σ in **Algorithm 2** is similar to sequential update proposed in Multi-Class CW [25]. The difference is that it solves the unconstrained optimization problem over each hypothesis, whereas the sequential update solves the constrained optimization problem. Also, **Algorithm 2** is an online learning algorithm in accordance with MIRA. However, our proposed approach can easily perform batch learning because it can be solved in closed form according to Eq. (11) and Eq. (13) instead of numerical computation for the quadratic programming problem.

6. Experiment and Result

We evaluated Structured AROW, which is our proposed approach, on various g2p tasks. The test sets in the g2p tasks include only unknown words because the correct pronunciation of known words can be obtained through dictionary lookup. First of all, we describe a preliminary experiment with a small dataset to investigate the characteristics of our proposed approach and determine various training parameters. Then, we describe the experiment to evaluate our proposed approach with various g2p tasks.

6.1 Preliminary Experiment

Table 1 shows datasets employed in the preliminary experiment; dataset name (Dataset), the number of grapheme and phoneme symbols (g/p), vocabulary sizes of training data (Train), development data (Dev), and test data (Test) and the number of trials of cross-validation (K-fold). Training, development, and test datasets are mutually exclusive. The development data is employed to determine various training parameters. In this experiment, we employ the NETtalk dataset, which is a small English dictionary obtained from the Pascal Letter-to-Phoneme Conversion Challenge[†]. We attempted to faithfully follow the convention in terms of data exclusion and data split in [13], except extracting development data from training data. To confirm that Structured AROW is robust to overfitting, we also create a separate Noisy NETtalk dataset, for which about 10% of the training data is artificial noisy data that has been given a wrong pronunciation randomly chosen from all pronunciations in NETtalk. That is, Noisy NETtalk includes 1760 noisy data of the total vocabulary size 17595. In Noisy NETtalk, the prediction performance of an approach that is not robust to overfitting can be expected to degrade by overfitting the noisy data.

Approaches evaluated in this experiment are the joint sequence model (Joint Sequence) which is the generative model employing joint n-grams for graphemes and phonemes, structured learning based on MIRA (MIRA), and Structured AROW which is our proposed approach. We employed Sequitur[††] as g2p conversion tool implementing the Joint Sequence and DirecTL+[†††] as g2p conversion tool implementing the MIRA. MIRA and our proposed approach employed context features, chain features, and joint n-gram features in accordance with [16]. The transition feature introduced in [16] was not used, as it was found to decrease performance in the NETtalk task. For alignment using in MIRA and our proposed approach, we used the unconstrained many-to-many alignment method of [7] as implemented in mpaligner[††††]. All discriminative methods employ phoneme error rate as their loss function. The context window size, joint n-gram size, hyperparameter r, n-best hypotheses for training, beam width for beam-search pruning, and training iterations were determined by phoneme error rate (PER) on the development data. Table 2 shows their details. The evaluation measures are PER, which indicates the rate of prediction errors on the phoneme level, and word error rate (WER), which indicates the rate of words for which the estimated pronunciation includes at least one phoneme error. Performances in speech recognition systems and text-to-speech systems depend on the WER. Also, the accuracy of a learning phoneme-level acoustic models when pronunciations are not given in a transcription depends on the PER. Also this experiment was performed on cluster machines equipped with Intel Xeon E5649 2.53GHz.

Table 3 shows the evaluation result on NETtalk and Noisy NETtalk. From the result of NETtalk in Table 3, it can be seen that the proposed approach and MIRA significantly outperformed Joint Sequence in terms of PER and WER. Compared with DirectTL+, our proposed approach has no significant difference in PER and WER. On the other hand,

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

[††] http://sequitur.info/

[†††] http://code.google.com/p/directl-p/

[††††] http://sourceforge.jp/projects/mpaligner/

Table 3 *Evaluation result in NETtalk and Noisy NETtalk. Values in this table are obtained by averaging results on each cross-validation. The best performance and performances that have no significant difference according to Paired Bootstrap Resampling [26] at a level of 0.05 over the best performance are written in bold.*

Dataset	Approach	PER (%)	WER (%)	Learning Time (hr.)
NETtalk	Joint Sequence	7.63	31.54	1.1
	MIRA	**6.75**	**28.15**	8.6
	Structured AROW	**6.75**	**28.56**	4.7
Noisy NETtalk	Joint Sequence	**9.78**	34.01	3.3
	MIRA	10.33	33.52	100.5
	Structured AROW	**9.79**	**33.02**	78.1

from the point of view of learning time, the learning speed of our proposed approach was faster than MIRA. Since our proposed approach calculates the closed forms only once for each hypothesis included in the n-best, the learning speed of our proposed approach is faster than structured learning based on MIRA, which has to iteratively seek the $\boldsymbol{w}$ that satisfies the constraints in Eq. (2) by a quadratic programming solver.

From the result of Noisy NETtalk in Table 3, the performance degradation of our proposed approach on noisy data is less than that of MIRA. The difference between our proposed approach and MIRA with regards to PER and WER is significant according to Paired Bootstrap Resampling [26] at a level of 0.05. As an example which show the significance of Structured AROW, the g2p conversions in a word "muffin" (pronunciation: mf̂xn) included in Noisy NETtalk are mentioned. Whereas Structured AROW correctly predicted it as "mf̂xn", MIRA incorrectly predicted it as "poltRgAstxn". The wrong pronunciation was estimated as "m"→"po", "u"→"l", "f"→"t", "f"→"RgAst", "i"→"x", and "n"→"n", where "m"→"po", "u"→"l", "f"→"t", and "f"→"RgAst" which can not found in NETtalk are wrong mappings between graphemes and phonemes generated by noisy data included in Noisy NETtalk. Although Structured AROW also deals with these wrong mappings, Structured AROW regarded them as the inappropriate prediction in a word "muffin". This result indicates that Structured AROW resolves MIRA's overfitting problems over noisy data, as it does for binary classification. On the other hand, although the fact that joint sequence model is also robust to overfitting in PER compared with MIRA was revealed, the PER is nearly the same as that of our proposed approach and the WER is signficantly higher than that of our proposed approach.

In Table 3, it can be noted that training time is significantly higher on Noisy NETtalk. This is because artificial noisy data included in Noisy NETtalk generates new and wrong mappings between graphemes and phonemes in the alignment step due to wrong pronunciations. The mappings increase the inferable pronunciation hypotheses $\hat{\boldsymbol{y}}$, and seriously affect time for predicting n-best hypotheses for discriminative training based on MIRA and Structured AROW. It also affects time for calculation of back-off smoothing on joint sequence model. However it is not a serious problem compared to that of the discriminative training methods. The problem on discriminative training can be controled by beam width in the beam-search pruning or solved using distributed training as proposed in [22].

6.2 Experiment with Various g2p Tasks

So far, we described two characteristics of Structured AROW through the preliminary experiment: the robust learning against overfitting, and the short learning time compared with that of MIRA. In the rest of this section, we describe an evaluation experiment for Structured AROW on various g2p tasks based on parameter settings of the preliminary experiment in order to explore the characteristics of the proposed algorithm in more detail.

Table 4 shows datasets employed in the experiment. Training, development, and test datasets are mutually exclusive. In this experiment, the development data is employed only to determine joint n-gram size, hyperparameter r, and the optimal number of training iterations. For datasets in Table 4, Brulex (French) and Beep (English) were obtained from the Pascal Letter-to-Phoneme Conversion Challenge as with NETtalk. CMUdict (English) and CELEX (English) were also obtained from their corresponding Web pages[†,††]. Wiktionary is a collaboratively constructed dictionary on the World Wide Web and is provided with archive data[†††]. We extracted words and pronunciations of American English written in International Phonetic Alphabet (IPA) from the provided archive data. Beep is also a collaboratively created dictionary derived from many sources and many people. The data preparation for the datasets followed [13] as with the preliminary experiment. For Wiktionary, the dataset extracted from the provided archive data included over two hundred phoneme symbols. We excluded data including minor phoneme symbols that appears less than 100 times. However it is still difficult task because there are still many phoneme symbols even after data cleaning is performed.

Table 5 shows parameter settings based on the preliminary experiment. On the preliminary experiment, the optimal value of joint n-gram was 5 or 7. For it, we attempted 5 and 7 in Joint Sequence, and 5 in MIRA and Structured AROW. The reason to only use joint 5-gram, which is shorter than joint 7-gram, in MIRA and Structured AROW is because they use many features not used

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

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

[†††] http://dumps.wikimedia.org/enwiktionary/

Table 6 *Evaluation result in various g2p tasks. Values on Brulex and CMUdict in this table are obtained by averaging results on each cross-validation. The best performance and performances that has no significant difference according to Paired Bootstrap Resampling [26] at a level of 0.05 over the best performance are written in bold.*

Dataset	Approach	PER (%)	WER (%)	Learning Time (hr.)
Brulex	Joint Sequence	1.27	6.61	3.8
	MIRA	**1.03**	**5.29**	3.3
	Structured AROW	1.08	5.59	2.4
CELEX English	Joint Sequence	2.62	12.15	4.1
	MIRA	**2.39**	**11.07**	29.4
	Structured AROW	2.51	11.81	15.1
CMUdict	Joint Sequence	6.77	28.55	17.5
	MIRA	**6.19**	**26.38**	55.4
	Structured AROW	**6.15**	**26.48**	28.5
Beep	Joint Sequence	2.26	12.24	32.0
	MIRA	2.35	12.60	238.9
	Structured AROW	**2.19**	**11.73**	255.2
Wiktionary	Joint Sequence	21.61	**60.91**	9.3
	MIRA	22.55	62.13	164.2
	Structured AROW	**21.23**	**60.19**	88.7

Table 4 *Datasets used in the experiment on the g2p task.*

Dataset	g/p	Vocabulary size			
		Train	Dev	Test	K-fold
Brulex	40/39	23353	1373	2747	5
CELEX English	26/53	39995	15000	5000	1
CMUdict	27/39	100886	5941	12000	2
Beep	26/44	169823	8938	19862	1
Wiktionary	26/87	63049	3709	7418	1

Table 5 *Parameter settings for the experiment. They were determined by the results of the preliminary experiment, except joint n-gram size, beam width, and hyperparameter r.*

	Joint Sequence	MIRA	Structured AROW
joint n-gram	5, 7	5	5
context window	-	6	6
n-best hypotheses	-	5	5
hyperparameter *r*	-	-	500, 1000, 1500
beam width	-	50	50

in Joint Sequence and thus can exceed its performance even with shorter n-grams. This improves efficiency in learning time and memory consumption. For Joint Sequence, joint 7-gram, which had the lowest PER in the development data of all datasets in Table 4, was finally selected as the joint n-gram size. Also we employed 50 beam width to decrease the computation cost because in NETtalk the g2p performance of 50 beam width was almost the same as that of 150 beam width. For hyperparameter *r*, the optimal value was different in each cross-validation fold on the preliminary experiment. Therefore we employ 500, 1000 and 1500 for it as with the preliminary experiment. This experiment was performed on same cluster machines as the preliminary experiment.

Table 6 shows the evaluation result on various g2p tasks. From Table 6, on Brulex and CELEX, MIRA significantly improved PER and WER compared with Structured AROW and Joint Sequence. On the other hand, on Beep and Wiktionary, Structured AROW improved PER and WER compared with MIRA with significant difference. Also, the PER of Structured AROW for these datasets was lower than that of Joint Sequence with significant difference. On CMUdict, the differences between Structured AROW and DirectTL+ for PER and WER are not significant.

We believe that the reason that MIRA has best performances on Brulex and CELEX is consistent rules for pronunciation annotation in these datasets. Because of this consistent annotation, the gap between the training data and the test data is small, then MIRA's overfitting problem did not surface. On the other hand, Structured AROW performed too much generalization over these datasets.

In contrast, it can be assumed that Beep and Wiktionary have various and inconsistent rules for pronunciation annotation because they created using the collective knowledge derived from many sources and many people. The rule from noisy data is also included therein. In this case, the gap between the training data and the test data is large, therefore Structured AROW obtained the best performance on these datasets by avoiding overfitting.

This result reveals that Structured AROW is suitable for learning from a dataset without a cross-check of language experts and can be applied to a difficult task having inconsistent rules. In contrast, MIRA is suitable for learning from a cross-checked clean dataset and applying to a easy task having consistent rules.

The learning time of Structured AROW was shorter than that of MIRA in almost datasets from Table 6, except Beep. This result is about the same as the preliminary experiment. However, in Beep, Structured AROW took more time for learning compared with MIRA. The reason is because in many cases, the $\ell_{h^2}(\boldsymbol{x}_i, \boldsymbol{y}_i, \hat{\boldsymbol{y}}_n, \boldsymbol{\mu}) = 0$ was not satisfied and thus many updates were performed. Such cases are caused by datasets including a large amount of data that many promising hypotheses (competitive candidates) can be estimated from. As a result, AROW saves learning time in many datasets compared with MIRA, although there are some exceptions.

For hyperparameter *r*, the high value 1500 was chosen on large datasets: CMUdict, Beep and Wiktionary. Also it tended to choose the low value 500 on small datasets: Brulex and CELEX. The optimal hyperparameter *r* is unknown and needs to be empirically chosen. As another solution, there are new adaptive algorithms [27] which allow AROW to adjust the hyperparameter *r* in each update. It can be considered as future work.

7. Conclusion

We proposed Structured AROW extending AROW to structured learning and evaluated it on various g2p tasks. In an evaluation experiment on dictionaries created using collective knowledge such as Beep and Wiktionary, our proposed approach significantly improved phoneme and word error rate compared with MIRA, by avoiding overfitting. Our proposed approach achieves a 6.8% error reduction rate compared to MIRA in terms of phoneme error rate on Beep. The result revealed that our proposed approach is more suitable than MIRA for datasets without a cross-check of language experts and for application to difficult tasks having inconsistent rules for pronunciation annotation. In addition, the learning speed of our proposed approach was faster than MIRA on the majority of datasets.

As future work, to further improve our proposed approach, we will consider an approach that approximately handles the covariance between two features in Σ within the limits of memory, or automatically adjusts *r*.

Acknowledgments

Part of this work was supported by JSPS KAKENHI Grant Number 24240032 and by the Commissioned Research of National Institute of Information and Communications Technology (NICT), Japan.

References

[1] L.R. Bahl, S. Das, P.V. Desouza, M. Epstein, R.L. Mercer, B. Merialdo, D. Nahamoo, M.A. Picheny, and J. Powell, "Automatic phonetic baseform determination," Proc. ICASSP, pp.173–176, 1991.

[2] J. Schroeter, A. Conkie, A. Syrdal, M. Beutnagel, M. Jilka, V. Strom, Y.J. Kim, H.G. Kang, and D. Kapilow, "A perspective on the next challenges for TTS research," Proc. 2002 IEEE Workshop on Speech Synthesis, 2002, pp.211–214, 2002.

[3] R.I. Damper, Y. Marchand, J.D. Marsters, and A.I. Bazin, "Aligning text and phonemes for speech technology applications using an EM-like algorithm," J. Speech Technology, vol.8, no.2, pp.147–160, 2005.

[4] S. Deligne, F. Yvon, and F. Bimbot, "Variable-length sequence matching for phonetic transcription using joint multigrams," Proc. EUROSPEECH, pp.2243–2246, 1995.

[5] S. Deligne and F. Bimbot, "Inference of variable-length linguistic and acoustic units by multigrams," Speech Commun., vol.23, no.3, pp.223–241, 1997.

[6] S. Jiampojamarn, G. Kondrak, and T. Sherif, "Applying many-to-many alignments and hidden Markov models to letter-to-phoneme conversion," Proc. NAACL HLT, pp.372–379, 2007.

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

[8] R.M. Kaplan and M. Kay, "Regular models of phonological rule systems," Computational linguistics, vol.20, pp.331–378, 1994.

[9] T.J. Sejnowski and C.R. Rosenberg, "Parallel networks that learn to pronounce English text," Complex Syst., vol.1, pp.145–168, 1987.

[10] W. Daelemans and A. van den Bosch, "Language-independent data-oriented grapheme-to-phoneme conversion," Progress in Speech Processing, pp.77–89, Springer-Verlag, 1997.

[11] S.F. Chen, "Conditional and joint models for grapheme-to-phoneme conversion," Proc. EUROSPEECH, pp.2033–2036, 2003.

[12] S. Deligne and F. Bimbot, "Inference of variable-length linguistic and acoustic units by multigrams," Speech Commun., vol.23, no.3, pp.223–241, 1997.

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

[14] K. Crammer and Y. Singer, "Ultraconservative online algorithms for multiclass problems," J. Machine Learning Research, vol.3, pp.951–991, 2003.

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

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

[17] T. Schlippe, S. Ochs, and T. Schultz, "Grapheme-to-phoneme model generation for Indo-European languages," Proc. ICASSP, pp.4801–4804, 2012.

[18] K. Crammer, A. Kulesza, and M. Dredze, "Adaptive regularization of weight vectors," Advances In Neural Information Processing Systems, pp.414–422, 2009.

[19] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," J. Machine Learning Research, vol.7, pp.551–585, 2006.

[20] 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.

[21] M. Dredze, K. Crammer, and F. Pereira, "Confidence-weighted linear classification," International Conference On Machine Learning (ICML), pp.264–271, 2008.

[22] K. Crammer, M. Dredze, and F. Pereira, "Confidence-weighted linear classification for text categorization," J. Machine Learning Research, vol.13, pp.1891–1926, 2012.

[23] R. Vanderbei, "Loqo: An interior point code for quadratic programming," Optimization methods and software, vol.11, no.1-4, pp.451–484, 1999.

[24] R. Zens and H. Ney., "Improvements in phrase-based statistical machine translation," Proc. NAACL HLT, pp.257–264, 2004.

[25] K. Crammer, M. Dredze, and A. Kulesza, "Multi-class confidence weighted algorithms," Empirical Methods in Natural Language Processing (EMNLP), pp.496–504, 2009.

[26] P. Koehn, "Statistical significance tests for machine translation evaluation," EMNLP, pp.388–395, 2004.

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

Keigo Kubo received his B.E. from Kinki University in 2009 and M.E. from the Graduate School of Information Science, NAIST, in 2011. He is currently the student of the doctoral program of the Graduate School of Information Science, NAIST. His research interests include structured learning for natural language processing such as g2p conversion, and speech recognition. He is a member of ISCA, and ASJ.

Sakriani Sakti received her B.E degree in Informatics (cum laude) from Bandung Institute of Technology, Indonesia, in 1999. In 2000, she received “DAAD-Siemens Program Asia 21st Century” Award to study in Communication Technology, University of Ulm, Germany, and received her MSc degree in 2002. During her thesis work, she worked with Speech Understanding Department, DaimlerChrysler Research Center, Ulm, Germany. Between 2003–2009, she worked as a researcher at ATR SLC Labs, Japan, and during 2006–2011, she worked as an expert researcher at NICT SLC Groups, Japan. While working with ATR-NICT, Japan, she continued her study (2005–2008) with Dialog Systems Group University of Ulm, Germany, and received her PhD degree in 2008. She actively involved in collaboration activities such as Asian Pacific Telecommunity Project (2003–2007), A-STAR and U-STAR (2006–2011). She also served as a visiting professor of Computer Science Department, University of Indonesia (UI) in 2009–2011. Currently, she is an assistant professor of the Augmented Human Communication Lab, NAIST, Japan. She is a member of JNS, SFN, ASJ, ISCA, IEICE amd IEEE. Her research interests include statistical pattern recognition, speech recognition, spoken language translation, cognitive communication, and graphical modeling framework.

Graham Neubig received his B.E. from University of Illinois, Urbana-Champaign, U.S.A, in 2005, and his M.E. and Ph.D. in informatics from Kyoto University, Kyoto, Japan in 2010 and 2012 respectively. He is currently an assistant professor at the Nara Institute of Science an Technology, Nara, Japan. His research interests include speech and natural language processing, with a focus on machine learning approaches for applications such as machine translation, speech recognition, and spoken dialog.

Tomoki Toda was born in Aichi, Japan on January 18, 1977. He earned his B.E. degree from Nagoya University, Aichi, Japan, in 1999 and his M.E. and D.E. degrees from the Graduate School of Information Science, NAIST, Nara, Japan, in 2001 and 2003, respectively. He was a Research Fellow of JSPS in the Graduate School of Engineering, Nagoya Institute of Technology, Aichi, Japan, from 2003 to 2005. He was an Assistant Professor of the Graduate School of Information Science, NAIST from 2005 to 2011, where he is currently an Associate Professor. He has also been a Visting Researcher at the NICT, Kyoto, Japan, since May 2006. From March 2001 to March 2003, he was an Intern Researcher at the ATR Spoken Language Communication Research Laboratories, Kyoto, Japan, and then he was a Visiting Researcher at the ATR until March 2006. He was also a Visiting Researcher at the Language Technologies Institute, CMU, Pittsburgh, USA, from October 2003 to September 2004 and at the Department of Engineering, University of Cambridge, Cambridge, UK, from March to August 2008. His research interests include statistical approaches to speech processing such as voice transformation, speech synthesis, speech analysis, speech production, and speech recognition. He received the 18th TELECOM System Technology Award for Students and the 23rd TELECOM System Technology Award from the TAF, the 2007 ISS Best Paper Award and the 2010 ISS Young Researcher’s Award in Speech Field from the IEICE, the 10th Ericsson Young Scientist Award from Nippon Ericsson K.K., the 4th Itakura Prize Innovative Young Researcher Award and the 26th Awaya Prize Young Researcher Award from the ASJ, the 2009 Young Author Best Paper Award from the IEEE SPS, the Best Paper Award (Short Paper in Regular Session Category) from APSIPA ASC 2012, the 2012 Kiyasu Special Industrial Achievement Award from the IPSJ, and the 2013 Best Paper Award (Speech Communication Journal) from EURASIP-ISCA. He was a member of the Speech and Language Technical Committee of the IEEE SPS from 2007 to 2009. He is a member of IEEE, ISCA, IPSJ, and ASJ.

Satoshi Nakamura received his B.S. from Kyoto Institute of Technology in 1981 and Ph.D. from Kyoto University in 1992. He was a director of ATR Spoken Language Communication Research Laboratories in 2000–2008, and a vice president of ATR in 2007–2008. He was a director general of Keihanna Research Laboratories, National Institute of Information and Communications Technology, Japan in 2009–2010. He is currently a professor and a director of Augmented Human Communication laboratory, Graduate School of Information Science at Nara Institute of Science and Technology. He is interested in modeling and systems of spoken dialog system, speech-to-speech translation. He is one of the leaders of speech-to-speech translation research projects including C-STAR, IWSLT and A-STAR. He headed the world first network-based commercial speech-to-speech translation service for 3-G mobile phones in 2007 and VoiceTra project for iPhone in 2010. He received LREC Antonio Zampoli Award, the Commendation for Science and Technology by the Ministry of Science and Technology in Japan. He is an elected board member of ISCA, International Speech Communication Association, and an elected member of IEEE SPS, speech and language TC.