Round-Robin Discrimination Model for Reranking ASR Hypotheses

Takanobu Oba, Takaaki Hori, Atsushi Nakamura

NTT Communication Science Laboratories, NTT Corporation.
{oba, hori, ats}@cslab.kecl.ntt.co.jp

Abstract

We propose a novel model training method for reranking problems. In our proposed approach, named the round-robin duel discrimination (R2D2), model training is done so that all pairs of samples can be distinguished from each other. The loss function of R2D2 for a log-linear model is concave. Therefore we can easily find the global optimum by using a simple parameter estimation method such as a gradient descent method. We also describe the relationships between the global conditional log-linear model (GCLM) and R2D2. R2D2 can be recognized as an expansion of GCLM. We evaluate R2D2 on an error correction language model for speech recognition. Our experimental results using the corpus of spontaneous Japanese show that R2D2 provides an accurate model with a high generalization ability.

Index Terms: discriminative training, reranking model, language model

1. Introduction

In general, a speech recognition result is a word sequence with the maximum score among many hypotheses generated by a speech recognizer. However, these hypotheses, in short a word n-best list or a lattice, include some word sequences with lower word error rates (WER) than the recognition result. One of the most typical ways to obtain such lower WER word sequences is the reranking/rescoring approach. Reranking is done by giving each hypothesis an extra score.

We focus on discriminative language models (DLM) based on a log-linear model [1, 2, 3, 4, 5, 6, 7, 8]. These approaches obtain pairs of reference and error words as a result of training using recognition hypotheses and is applied to reranking. So DLM based reranking is often referred to as an error correction approach. The parameter estimation is formalized as a minimization problem of a predefined loss function. The purpose of training is to give large scores to word sequences with low WERs, and to give small scores to the others.

Some training methods for the reranking problem have been proposed in the machine learning area. They include reranking boosting (ReBst) [9], the global condition log-linear model (GCLM) [1, 2, 6, 10] and minimum error rate training (MERT) [11]. For better parameter estimation we often use sample weights, where the term ‘sample’ means a word sequence in a word n-best list or lattice, in short, a hypothesis, and the sample weights indicate the importance of the word sequence. Typically WER is used as a sample weight for DLM training. So we can recognize samples with small weights as approximated references.

In this paper, we propose a novel model training method for the reranking problem. Our proposed method employs a sample weight based training. The loss function is designed so that all pairs of samples are distinguished from each other. All samples take turns to be references and to be competitors.

An important characteristic of the proposed method, the round-robin duel discrimination (R2D2) approach, is the convexity of the loss function. So we can easily find the global optimum by using a normal parameter estimation method such as the gradient descent method and the quasi-Newton method. We briefly prove the convexity of the loss function in this paper. In addition, R2D2 can be viewed as an expansion of GCLM. We describe the mathematical relationship between these two models.

One of problems of GCLM is the risk of overfitting. The loss function of GCLM is designed so that the reference sample is distinguished from the other samples. As a result, models trained by GCLM tend to be tinged with features of the reference more strongly than those of the others. Hence, such models provide low accuracy for mismatched data, although GCLM would provide an accurate model if we could prepare a large amount of training data. R2D2 distinguishes between all the samples, in other words, several samples with relatively small sample weights are distinguished from the other samples. This mitigates the overfitting risk.

The loss function of MERT is also constructed so that some samples have positive scores as well as R2D2. Hence, MERT would provide a data-robust model. However, there is no assurance that MERT gives large scores to samples with small sample weights. Therefore, R2D2 outperforms MERT for matched data.

We compare R2D2 with some conventional reranking model training methods using the corpus of spontaneous Japanese (CSJ) [12]. We construct DLMs and rerank the 5000-best hypotheses of speech recognition. Our experimental results show that R2D2 provides an accurate model with high generalization ability.

This paper is organized as follows: in section 2, we describe the basics of n-best hypotheses reranking and model training. R2D2 is proposed and its characteristics are described in section 3. Section 4 provides our experimental results. And section 5 concludes this paper.

2. Reranking Hypotheses

We represent n-best hypotheses generated from a speech recognition system as \( L = \{ h_j | j = 1, 2, \ldots, N \} \), and a feature vector of a hypothesis \( h \) as \( f(h) \). Specifically, the speech recognition score of \( h \) is denoted as \( f_0(h) \).

Where \( a \) is a given parameter vector, the goal of the reranking problem on the speech recognition is to find a hypothesis with the highest score. This is formulated as follows.

\[
    h^* = \arg \max_{h \in L} \{ a_0 f_0(h) + a^T f(h) \} \quad (1)
\]
where \( a_0 \) is a given scaling constant. \( \top \) denotes the transpose of the matrix.

For training, we prepare a data set that comprises:

- N-best lists \( \{ L_i | i = 1, 2, \ldots, I \} \)
- They are generated from a speech recognizer for training data consisting of \( I \) utterances. Each hypothesis is converted to a feature vector, which is denoted as \( f_{i,j} \). That is, \( L_i = \{ f_{i,j} | j = 1, 2, \ldots, N_i \} \).
- WER as sample weight
  The WER of each hypothesis \( e_{i,j} \) is used as a sample weight for training.

The parameters are estimated by finding a parameter vector \( \mathbf{a} \) that minimizes a predefined loss function. For example, the loss functions of ReBst and weighted GCLM (WGCLM) [10] are as shown below.

\[
\mathcal{L}_{\text{ReBst}} = \sum_{i=1}^{I} \sum_{j=1}^{N_i} e_{i,j} \exp(\mathbf{a}^\top f_{i,j}) \exp(\mathbf{a}^\top f_{i,r})
\]
\[
\mathcal{L}_{\text{WGCLM}} = \sum_{i=1}^{I} \log \left( \sum_{j=1}^{N_i} e_{i,j} \exp(\mathbf{a}^\top f_{i,j}) \right)
\]

3. Round-Robin Duel Discrimination

3.1. Loss of R2D2

The loss function of our proposed method, R2D2, is defined as follows.

\[
\mathcal{L}_{\text{R2D2}} = \sum_{i=1}^{I} \log \left( \sum_{j=1}^{N_i} \frac{e_{i,j} \exp(\mathbf{a}^\top f_{i,j}) \exp(\mathbf{a}^\top f_{i,j}')} \right)
\]

In each i-th list, all pairs of samples are considered and distinguished from each other taking the sample weights into account. Although \( \bar{e} \) and \( \bar{e} \) are typically sample weights, we introduce \( \bar{e}_{i,j} = \exp(\sigma_1 e_{i,j}) \) and \( \bar{e}_{i,j}' = \exp(\sigma_2 e_{i,j}') \) where \( \sigma_1 \) and \( \sigma_2 \) are hyperparameters. One reason for this is to avoid a 0 denominator. Also, it arises from the success of the exponential weight in acoustic model training, e.g., boosted maximum mutual information. Furthermore, it makes it easy to explain the relationships between R2D2 and GCLM.

Next, we focus on how to calculate a summation over \( \sum_{j=1}^{N_i} \sum_{j'=1}^{N_i} \), since direct calculation is computationally expensive if \( N_i \) is large. We can reduce an \( O(N_i^2) \) order computation to an \( O(N_i) \) order computation by calculating the numerator and the inverse of the denominator separately. That is, first we calculate

\[
n_i = \sum_{j=1}^{N_i} \exp(\mathbf{a}^\top f_{i,j}) \bar{e}_{i,j}
\]
\[
d_i = \frac{1}{\sum_{j'=1}^{N_i} \exp(\mathbf{a}^\top f_{i,j'})} \bar{e}_{i,j'}
\]

and then we multiply these two terms as

\[
\sum_{j'=1}^{N_i} \sum_{j=1}^{N_i} \bar{e}_{i,j} \exp(\mathbf{a}^\top f_{i,j'}) \exp(\mathbf{a}^\top f_{i,j'}) = n_i \cdot d_i.
\]

3.2. Concavity

The loss function of R2D2 is concave because it satisfies

\[
\sum_{j'=1}^{N_i} \sum_{j=1}^{N_i} \bar{e}_{i,j} \exp(\mathbf{a}^\top f_{i,j'}) \exp(\mathbf{a}^\top f_{i,j'}) > 0
\]

\[
\sum_{j'=1}^{N_i} \sum_{j=1}^{N_i} \bar{e}_{i,j} \exp(\mathbf{a}^\top f_{i,j'}) \exp(\mathbf{a}^\top f_{i,j'}) > \sum_{j'=1}^{N_i} \sum_{j=1}^{N_i} \bar{e}_{i,j} \exp(\mathbf{a}^\top f_{i,j'}) \exp(\mathbf{a}^\top f_{i,j'})
\]

where \( R \) is the number of samples with 0 weight. Consequently, \( \mathcal{L}_{\text{R2D2}} \) corresponds to the loss function of GCLM when \( R = 1 \). Note that we use the Oracle reference and set its WER to be 0 if the n-best list contains no hypothesis with no error.
3.4. Construction of loss function denominators

Figure 1 shows discrimination images of WGCLM, R2D2 and MERT. Training aims to give large scores to the samples in the right boxes, which correspond to the denominators of the loss functions. Clearly distinguished samples are denoted by heavy bold arrows.

The denominator of the loss of WGCLM consists of a single sample, viz., a reference of an n-best list. As a result of training, the reference features have much larger scores than those of the other samples. A model that gives large scores only to a few samples poses an overfitting risk.

In R2D2, the denominator is constructed using sample weights. If $\bar{e}_{i,j}$ is small, the corresponding sample is clearly distinguished from the others, because $\bar{e}_{i,j}^\alpha$ become large. Hence, several samples with relatively small sample weights have large scores. Therefore, R2D2 can mitigate the overfitting risk better than WGCLM.

The loss function of MERT is

$$L_{\text{MERT}} = \sum_{i=1}^{N} \sum_{j=1}^{N'} e_{i,j} \exp(a^\top f_{i,j})^\alpha$$

(18)

where $\alpha$ is a hyperparameter. Since the denominator consists of several samples, MERT can also mitigate the overfitting risk. However, there is no assurance that MERT gives large scores to samples with small sample weights. Consequently, R2D2 outperforms MERT with matched data.

4. Experiments

4.1. Experimental conditions

We used CSJ for our experiments. CSJ includes many lectures and their transcriptions. The lectures consist of academic and simulated presentations.

Table 1 shows the amount of data in our experimental environment. To make 5000-best lists, the utterances were recognized by using the speech recognition system SOLON, which was developed at NTT CSLabs. SOLON is a decoder based on a weighted finite state transducer and it can provide a fast efficient search by using a fast on-the-fly composition algorithm [15]. The acoustic model consists of MCE trained tri-phone HMMs with 5,000 states and 32 Gaussians [16]. The language model is a tri-gram model with Kneser-Ney smoothing.

Test set 2 and its development set consist of simulated lectures while the others consist of academic lectures. Test-set perplexities were calculated by using the language model of the SOLON speech recognition system. The perplexities of the training set and test set 1 are very similar. We can expect test set 1 to include very similar linguistic features to the training set. Test set 2 includes very different features, since its perplexity is much larger than the others. Thus, we consider test sets 1 and 2 to be matched and mismatched conditions, respectively.

To make a model, we introduce L-2 norm for regularization. The objective function is substituted with

$$\mathcal{L}_a + \frac{||a||}{C}$$

(19)

We use the L-BFGS algorithm [17] for parameter estimation, except for ReBst, for which a dedicated algorithm is applied. Each development set is used to decide the scaling constant $a_0$ in equation (1), $\alpha$ in MERT, the regularization constant $C$ and the convergence conditions in ReBst. The model selected using the development set is applied for the corresponding test set.

Feature vectors consist of word uni-, bi-, tri-gram booleans. A boolean feature has the value of 0 if the corresponding n-gram entry dose not occur in a sentence, 1 otherwise.

Table 1: Experimental data. Dev. lect., uttr. and p.p. denote development, lectures, utterances and perplexity, respectively.

<table>
<thead>
<tr>
<th></th>
<th>lecture</th>
<th># of lect. (uttr.)</th>
<th># of words</th>
<th>p.p.</th>
</tr>
</thead>
<tbody>
<tr>
<td>training</td>
<td>academic</td>
<td>150 (25, 130)</td>
<td>420, 918</td>
<td>68.7</td>
</tr>
<tr>
<td>dev. set</td>
<td>academic</td>
<td>10 (1, 293)</td>
<td>26, 329</td>
<td>76.1</td>
</tr>
<tr>
<td>test set</td>
<td>academic</td>
<td>10 (1, 156)</td>
<td>26, 798</td>
<td>74.4</td>
</tr>
<tr>
<td>dev. set</td>
<td>simulated</td>
<td>10 (1, 479)</td>
<td>20, 996</td>
<td>96.1</td>
</tr>
<tr>
<td>test set</td>
<td>simulated</td>
<td>10 (717)</td>
<td>17, 242</td>
<td>142.3</td>
</tr>
</tbody>
</table>
Table 2: WERs before/after applying reranking models.

<table>
<thead>
<tr>
<th></th>
<th>test set 1</th>
<th>test set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before reranking</td>
<td>18.0</td>
<td>34.5</td>
</tr>
<tr>
<td>ReBst</td>
<td>17.8</td>
<td>33.7</td>
</tr>
<tr>
<td>MERT</td>
<td>17.7</td>
<td>33.1</td>
</tr>
<tr>
<td>WGCLM</td>
<td>17.4</td>
<td>33.3</td>
</tr>
<tr>
<td>R2D2</td>
<td>17.2</td>
<td><strong>32.9</strong></td>
</tr>
</tbody>
</table>

Table 3: WERs obtained using several models trained by R2D2 giving different hyper parameter values for exponential weights.

<table>
<thead>
<tr>
<th>σ₁, σ₂</th>
<th>dev set 1</th>
<th>test set 1</th>
<th>dev set 2</th>
<th>test set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1 0.5</td>
<td>19.9</td>
<td>17.3</td>
<td>35.2</td>
<td>33.0</td>
</tr>
<tr>
<td>0.5 0.5</td>
<td>19.8</td>
<td>17.2</td>
<td>35.3</td>
<td>32.9</td>
</tr>
<tr>
<td>0.5 2.0</td>
<td>19.8</td>
<td>17.1</td>
<td><strong>35.0</strong></td>
<td><strong>32.9</strong></td>
</tr>
<tr>
<td>1.0 2.0</td>
<td>19.8</td>
<td>17.2</td>
<td>35.1</td>
<td>32.9</td>
</tr>
<tr>
<td>2.0 2.0</td>
<td><strong>19.7</strong></td>
<td><strong>17.2</strong></td>
<td>35.2</td>
<td>33.0</td>
</tr>
<tr>
<td>0.1 ∞</td>
<td>20.0</td>
<td>17.2</td>
<td>35.2</td>
<td>33.5</td>
</tr>
<tr>
<td>0.5 ∞</td>
<td>19.8</td>
<td>17.0</td>
<td>35.1</td>
<td>33.2</td>
</tr>
<tr>
<td>1.0 ∞</td>
<td>19.9</td>
<td>17.1</td>
<td>35.2</td>
<td>33.1</td>
</tr>
<tr>
<td>2.0 ∞</td>
<td>19.8</td>
<td>17.1</td>
<td>35.1</td>
<td>33.2</td>
</tr>
</tbody>
</table>

4.2. Results

Table 2 shows the WERs before and after applying the reranking models. Before reranking means the results of 1-best generated from SOLON. When the absolute differences of the WERs are 0.2 and 0.3 for test sets 1 and 2, respectively, they are statistically significant (p < 0.02).

The model trained by ReBst slightly reduces the number of recognition errors. When comparing MERT and WGCLM, WGCLM outperforms MERT for test set 1. As mentioned in section 3.4, MERT does not directly give large scores to word sequences with low WERs. Hence, WGCLM provides better results than MERT under matched conditions, in which the training and evaluation data are linguistically similar. Of the four methods, R2D2 provides the most accurate model. The differences are significant for both test sets 1 and 2 against WGCLM.

Table 3 shows relationships between WER and (σ₁, σ₂). All the hyperparameters, except for σ₁ and σ₂, are adjusted to each development set and the model is applied to the corresponding test set. The results shown in bold type are the same as in table 2.

R2D2 performs robustly against σ₁ and σ₂ if σ₂ ≠ ∞. The models under σ₂ = ∞ slightly underperform R2D2 with σ₂ ≠ ∞ for test set 2. σ₂ = ∞ provides the loss function of equation (17) with ɛᵢᵢ’s. Since the denominators consist only of word sequences with WER=0, the overfitting risk is higher than R2D2 with σ₂ ≠ ∞. However, it was not so serious because most n-best lists had multiple word sequences with WER=0 in our experimental data.

In contrast, R2D2 with σ₂ = ∞ accurately worked for test set 1. This result depends on the number of n-best lists that have multiple word sequences with WER=0 and the number of such word sequences. If most n-best lists would have only one word sequence with WER=0, the accuracy would be lower for test set 1 since such condition just corresponds to WGCLM.

5. Conclusion

We proposed a novel model training method, named R2D2. An important feature of R2D2 is the concavity of the loss function. Hence, we can easily find the global optimum by using a normal parameter estimation method such as the quasi-Newton method. In addition, we showed the relationship with GCLM. The loss function of GCLM is derived by considering the limits of the R2D2 hyperparameter. Our experimental results also revealed high generalization ability of R2D2. R2D2 outperformed conventional methods and provided an accurate model for both matched and mismatched conditions.

6. References