Empirical Sampling from Latent Utterance-wise Evidence Model for Missing Data ASR based on Neural Encoder-Decoder Model

Ryu Takeda1, Yui Sudo2, Kazuhiro Nakadai2,3 and Kazunori Komatani1

1SANKEN, Osaka University, Japan
2Honda Research Institute Japan, Co., Ltd., Japan
3Tokyo Institute of Technology, Japan

rtakeda@sanken.osaka-u.ac.jp, yui.sudo@jp.honda-ri.com, nakadai@ra.sc.e.titech.ac.jp, komatani@sanken.osaka-u.ac.jp

Abstract

Missing data automatic speech recognition (MD-ASR) can utilize the uncertainty of speech enhancement (SE) results without re-training of model parameters. Such uncertainty is represented by a probabilistic evidence model, and the design and the expectation calculation of it are important. Two problems arise in applying the MD approach to utterance-wise ASR based on neural encoder-decoder model: the high-dimensionality of an utterance-wise evidence model and the discontinuity among frames of generated samples in approximating the expectation with Monte-Carlo method. We propose new utterance-wise evidence models using a latent variable and an empirical method for sampling from them. The space of our latent model is restricted by simpler conditional probability density functions (pdfs) given the latent variable, which enables us to generate samples from the low-dimensional space in deterministic or stochastic way. Because the variable also works as a common smoothing parameter among simple pdfs, the generated samples are continuous among frames, which improves the ASR performance unlike frame-wise models. The uncertainty from a neural SE is also used as a component in our mixture pdf models. Experiments showed that the character error rate of the enhanced speech was further improved by 2.5 points on average with our MD-ASR using transformer model.

Index Terms: missing data, end-to-end ASR, evidence model, speech enhancement

1. Introduction

Integration techniques between automatic speech recognition (ASR) and frontend signal processing, such as speech enhancement (SE) or source separation (SS), are important for expanding the application areas and users of ASR. We can easily try state-of-the-art ASR or SE/SS based on deep neural networks (DNNs) thanks to the programming libraries [1], frameworks, and sometimes provided pre-trained models [2, 3]. However, the training cost of them, such as the data and computer resources, increases as DNNs become more complex like ASR models. Such high-cost joint (re-)training of ASR and SE/SS models is required to achieve high performance whenever we change one of the models to another for a different application and acoustic environment. We believe that a joint-training-free interface between ASR and SE/SS will make each model more flexible and portable for real applications.

The recent integrations of ASR and SE/SS are based on a cascaded-connection of them [4, 5] or on end-to-end models and optimization in the neural framework [6, 7, 8]. Frontend SE/SS improves the performance of ASR for target speech by reducing the influence of other signals, such as environmental sounds and non-target speech. Although a cascaded connection of SE and ASR improves the recognition performance, the performance of end-to-end optimization using large data is better. On the other hand, such optimization generally has the trade-off between the performance and the specialization to a specific data set.

Another direction is the missing data ASR (MD-ASR) approach, which does not necessarily require the re-training of models. This approach utilizes the uncertainty of speech features in the decoding process [9, 10, 11], which can propagate statistical information from SE/SS to MD-ASR [12, 13, 14, 15]. The probabilistic model of such uncertainty is called an evidence model [11, 16, 17] represented as probability density function (pdf) which reduces misclassification risks by evaluating the expectation of target values, such as the classification score. Note that the formulation of the evidence model and MD-ASR usually depends on the structure of the base ASR, such as hidden Markov model (HMM) with a Gaussian mixture model (GMM) [11, 16] or DNN-HMM [18, 19, 20, 21]. For example, the frame-wise or element-wise evidence models have been designed in accordance with their acoustic likelihood or posterior probability defined at each frame. Although uncertainty information can be used as one of the input features or data samples in the training phase [22, 23], we should not increase mutual dependence of model parameters between specific ASR and SE/SS for the portability of each model and method.

Two problems arise in applying the MD approach to recent neural ASR based on encoder-decoder (ED) architectures [24, 25, 26, 27]: the high dimensionality of an utterance-wise evidence model and the generation of discontinuous speech features from it (Fig. 1). Because ED-ASR recognizes speech in utterances not frame-by-frame, the straightforward utterance-wise model becomes a function of the sequence of speech features. Although Monte-Carlo approximation for expectation calculation is a realistic strategy for highly-nonlinear neural ASRs, sample generation from such models is difficult due to its high dimensionality. The continuity as a sequential feature is also important for the generated samples. The previous frame-wise evidence models such as [11] may generate unnatural and discontinuous features because these models assume the independence among frames, which results in recognition failure.

We propose new utterance-wise evidence models conditioned by a latent variable and an empirical method for sam-
pling from the models. The space of our basic utterance-wise model is restricted by simpler conditional pdfs given the latent variable, which enables us to draw samples from a substantially low-dimensional space. Because the latent variable also works as a common smoothing parameter among each simple pdf, the generated samples from them are continuous and smooth among frames. The uncertainty pdf from a neural SE in the feature-domain is used as a mixture component of our evidence model. The samples are empirically selected deterministically or stochastically in accordance with each pdf in our models. We also examine which variables of ED-ASR are appropriate for the expectation calculation, hidden vectors or probabilities.

2. Preliminary

Our target ASR architecture and the principle of the evidence model are explained along with the decoding process by showing specific models.

2.1. Encoder-Decoder-based Neural ASR

End-to-end ASR directly estimates a character sequence $c_{1:T} = [c_1, ..., c_T]$ with length $T$ from a sequence of speech feature vectors $s_{1:T} = [s_1, ..., s_T]$ with length $T$. The feature vector at frame $t$, $s_t \in \mathbb{R}^D$, is a $D$-dimensional vector. The mapping from $s_{1:T}$ to $c_{1:T}$, e.g., the posterior probability $p(c_{1:T}|s_{1:T})$, is modeled by DNNs and obtained by supervised training.

We focus on the hybrid CTC/attention architecture [27] as a recent standard end-to-end ASR. This architecture consists of a shared encoder, CTC and attention decoder networks. The shared encoder network converts speech features $s_{1:T}$ into a $T'$-length sequence of encoded hidden vectors $h_{1:T'} = [h_1, ..., h_{T'}]$ with $h_t \in \mathbb{R}^D$ ($t = 1, ..., T'$). The CTC and attention decoder networks represent posterior probabilities $p_{ctc}(c_{1:T}|h_{1:T'})$ and $p_{att}(c_{1:T}|h_{1:T'})$ in different ways. The score function $J_{att}$ in the decoding process is represented by the weighted sum of them with weights $0 \leq w_l \leq 1$, $(l = 1, 2)$. These processes are simply summarized as follows.

$$J_{att} = \log p_{ctc}(c_{1:T}|h_{1:T'}) + w_2 \log p_{att}(c_{1:T}|h_{1:T'})$$

Each function, Encoder, $p_{ctc}$, and $p_{att}$, may be implemented with recurrent neural networks, transformers or conformers.

2.2. Frame/Element-wise MD-ASR and Evidence Model

MD-ASR considers the uncertainty of speech features $s_{1:T}$ caused by noise or remaining distortion after SE. The uncertainty of $s_{1:T}$ is represented by an evidence model (pdf) with a parameter set including $s_{1:T}$ itself.

We review the principle of the evidence model in the case of HMM models without loss of generality. MD-ASR based on HMM evaluates the $t$-th frame acoustic likelihood $p(m_t|s_t)$ at a hidden state $m_t$ in the decoding process. The frame-wise evidence model is a function of the $t$-th feature vector $s_t$ with a parameter set $\theta_t$. If independence among features of vector is assumed, the element-wise evidence model becomes a function of $s_{1:T}$ with a parameter set $\theta_{1:T}$, where $d$ is a dimension index of $s$.

The expectation of target functions or variables is evaluated using the evidence model. The acoustic likelihood of the $d$-th dimension of feature vector $s_t$ is marginalized to evaluate the expectation by using the following equation:

$$\mathbb{E}[p(s_{1:T}|m_{1:t})] := \int p(z|m_t)E(z; \theta_{1:t}) dz,$$

where $s_{1:d}$ appears as a realization of $\theta_{1:d}$ in the integral. The computation of the expectation integral depends on the model. For example, the expectation is calculated analytically if both $p(z|m_t)$ and $E(z; \theta_{1:t})$ are a Gaussian pdf. Sample-based approximation of the expectation, such as unscented transformation [28] or the Monte-Carlo method, can be applied otherwise.

Several evidence models for $E(z; \theta_{1:t})$ have been proposed for frame-wise MD-ASRs [11, 16]. In particular, the Dirac-delta, Uniform, and Gaussian pdfs or a mixture of them are widely used. For example, the mixture pdf of Dirac-delta and Uniform pdfs with a mixture weight $r$ is expressed as

$$E(z; \theta_{1:t}) = \pi \delta(z; z_m) + (1 - \pi)U(z; z_m, z_m),$$

where $z_m, z_m > z_m$ specify the distribution boundaries, and $\pi$ is an indicator function equal to 1 if $z \in [z_m, z_m]$, and 0 otherwise. $z_m$ specifies a specific value. The evidence $E_L(z; \theta_{1:t})$ means that $z$ takes $z_m$ with probability $\pi$, and $z$ may be in the range of $[z_m, z_m]$ with probability $(1 - \pi)$. These parameters define the distribution of $s_{1:T}$.

The value of speech feature $s_{1:T}$ is usually set to $z_m$. The feature value of the observed raw noisy signal is set to $z_u$ in the mel-filterbank case. $z_u$ is set to the minimum flooring value of the feature.

3. Proposed Method

3.1. Utterance-wise Evidence Model in Neural ED-ASR

Our targets for the expectation calculation using the evidence model are the encoded vectors $h_{1:T'}$ and the posterior probability $p_{ctc}$ in the ED-ASR. These two can be evaluated independently from the search process while $p_{att}$ is not due to the search context. The use of the expected $h_{1:T'}$ has the potential to reduce classification errors for $p_{ctc}$ and $p_{att}$ thanks to the shared architecture.

The expectation of $h_{1:T'}$ and $p_{ctc}$ is formulated based on the basis of an utterance-wise evidence model $E$ as follows:

$$E[h_{1:T'}(s_{1:T})] := \int h_{1:T'}(z_{1:T})E(z_{1:T}; \Theta)dz_{1:T},$$

$$E[p_{ctc}(c_{1:T}|h_{1:T'})] := \int p_{ctc}(c_{1:T}|h_{1:T'}(z_{1:T}))E(z_{1:T}; \Theta)dz_{1:T},$$

where $s_{1:T}$ becomes a realization of the parameter set $\Theta$. Because it is difficult to calculate the integral over the speech features $s_{1:T}$ through highly non-linear networks analytically, we use a Monte-Carlo-like approximation of them by using $N$ samples $\{s_{1:T}^{(n)}\}_{n=1}^N$ drawn from $E(z_{1:T}; \Theta)$. The left side of Fig. 2 shows our implementation in accordance with Eqs. (6) and (7).

The key to the sample-based approach is how to select good samples from the obviously high-dimensional space $s_{1:T} \in \mathbb{R}^D \times T$. Here, frame-wise models will not generate continuous samples due to the independence among frames. Note that the generated samples can be processed in parallel as a GPU minibatch in the decoding process. The mean operation of them approximates the expectation operator in Eqs. (6) and (7).
3.2. Latent Space Model and Empirical Sampling

Our idea is to assume a low-dimensional space for the utterance-wise evidence model and select samples from the simple space. By introducing a latent variable into the evidence model, we can decompose a full utterance-wise evidence model into a set of simple conditional evidence models. The power-spectrum-based feature, such as mel-filterbank, is assumed hereafter.

First, we design a basic and stable utterance-wise Uniform pdf $E_{t,d}$ for speech feature $s_{t,d}$ to satisfy the sequential continuity. $E_{t,d}$ corresponds to $E_{t}$ in Eq. (5). Here, we denote the sequential features extracted from noisy observed signals as $u_{t}=\{u_{1}, ..., u_{T}\}$, $u_{t}\in\mathbb{R}^{D}(t=1, ..., T)$. By assuming conditional independence between $z_{t,d}$ and a latent variable $\alpha$ as shown in Fig. 2, we decompose $E_{t}$ as

$$E_{t}(z_{t}; \Theta) = \int p(z_{t}|\alpha) p(\alpha) d\alpha,$$

$$= \int \prod_{d=1}^{D} E_{t,d}(z_{t,d}; (1-\alpha)s_{t,d} + \alpha u_{t,d})|E_{t}(\alpha; 0, 1)dx_{t},$$

where $u_{t,d}$ is the value of the d-th dimension of $u_{t}$. $E_{t,d}$ and $E_{t}$ are defined in Eq. (5). We can draw sequentially-continuous samples from this evidence model easily by applying ancestral sampling; draw $\alpha$ from the uniform distribution and determine $z_{t,d}$ in accordance with the Delta function. Since $\alpha$ follows the Uniform pdf, we can deterministically select samples of $\alpha$ at even intervals in $[0, 1]$ with an arbitrary resolution.

Next, we introduce the uncertainty from SE into the evidence model. Here, we assume that SE provides a frame-wise posterior probability $p_{se}(s_{t})$ for MD-ASR. We define the mixture pdf of $E_{t}(s_{t})$ and the posterior as

$$E_{t}(z_{t}; \Theta) = \pi p_{se}(s_{t}) + (1-\pi)E_{t}(z_{t}; \Theta),$$

where $\pi$ is a mixture weight. The Uniform evidence model works as a probabilistic flooring when the accuracy of $p_{se}$ is not good due to a model mismatch. A single posterior and a mixture pdf of Uniform and Delta can also be evidence models.

$$E(z_{t}; \Theta) = \pi \prod_{d=1}^{D} E_{t,d}(z_{t,d}; s_{t,d}) + (1-\pi)E_{t}(z_{t}; \Theta),$$

$$E_{t,d}(z_{t,d}; \Theta) = \prod_{u_{t,d}} p_{se}(u_{t,d}),$$

The differences among these evidence models are examined experimentally.

The sampling from our evidence models with the posterior probability $p_{se}$ from SE includes a stochastic process. We allocate the number of available samples drawn from each pdf to exclude the stochastic process as far as possible in advance. For example, if the total number of available samples is $N$, we divide $N$ samples into $n N$ and $(1-\pi)N$ for mixture models with a weight $\pi$ in Eqs. (10) and (11). The samples from $E_{t}$ are selected deterministically, and those from $p_{se}$ are selected stochastically in the case of Eq. (10).

3.3. Posterior of Monaural Neural SE in Feature Domain

We represent the uncertainty $p_{se}$ of $s_{t}$ as a posterior probability given an observed noisy signal with DNNs. Although SE itself estimates the $F$-dimensional denoised spectrum $y_{t}$, $t=\{y_{1}, ..., y_{T}\}$, $y_{t}\in C^{F}(t=1, ..., T)$ from the single-channel noisy observed spectrum $x_{t}$, $t=\{x_{1}, ..., x_{T}\}$, $x_{t}\in C^{D}(t=1, ..., T)$ in the short-time Fourier transform (STFT) domain, our $p_{se}$ is designed for the feature domain. Here, we assume that the frame index $t$ and length $L$ of sequential variables both in the STFT and feature domains is same. We also assume that the feature $s_{t}$ of the denoised $y_{t}$ is obtained through deterministic feature-extraction function $f$.

Mask-based SE estimates $y_{t}$ by multiplying soft masks by the observed spectrum $x_{t}$ [29], i.e., $y_{t} = m_{t} \circ x_{t}$. Here, $\circ$ represents element-wise multiplication. The soft-mask vector $m_{t}$ is a function of $x_{t}$, $t$ or its subset modeled by DNNs, and each element of $m_{t}$ is in the range $[0, 1]$. By assuming that prediction errors $n_{t}^{m}$ and $n_{t}^{f}$ follow Gaussian pdf, we model the process from $x_{t}$ to $s_{t}$ and its probability as

$$y_{t} = \log(m_{t}(x_{t}) \circ |x_{t}|) + n_{t}^{m} = N(0, \Lambda_{m}^{-1}),$$

$$s_{t} = f(y_{t}) + n_{t}^{f},$$

where $\log(\cdot)$ and $|\cdot|$ represent logarithm and absolute functions. $x_{t}$ is a concatenated vector of $x_{t}$ from frame $t-k$ to $t+k$, $A_{x,t}$ and $A_{x,t}$ are precision matrices, and they are estimated also as functions of $x_{t}$ modeled by DNNs. We assume that $A_{y,t} = \Lambda_{y,t} = \Lambda_{y} = \Lambda_{y}L_{y}^{T}$ [10].

The ideal training of the DNNs is based on a lower-bound of the log-likelihood of $p(s_{t}|x_{t}) = \int p(s_{t}|y_{t}, x_{t})p(y_{t}|x_{t})dy_{t}$, defined from Eqs. (13) – (14), and the lower-bound becomes

$$\log p(s_{t}|x_{t}) \geq E_{t} \log p(y_{t}|x_{t})/q(y_{t}|x_{t}),$$

where $y_{t}$ equals $\log|y_{t}|$, and the second term just promotes the training of $m_{t}(x_{t})$. The posterior probability $p(s_{t}|y_{t}, x_{t}) = N(f(y_{t}), \Lambda_{t}^{-1}(y_{t}, x_{t}))$ is used as $p_{se}$ in Eqs. (10) and (12) instead of the original log-likelihood.

4. Experiment

4.1. Experimental Setups

i) Data Set for ASR and Speech Enhancement

We conducted experiments with speech data from the Corpus of Spontaneous Japanese (CSJ) [32] and non-speech data from the ProSoundEffect (PSE) corpus that was also used in [33]. The training set data contained about 230 hours of academic lecture presentations, and the official evaluation sets of CSJ (eval1, eval2, and eval3: 5 hours in total) were used for test sets. Each piece of speech data was segmented into utterances. The sampling rate of all data was set to 16K Hz. The PSE corpus includes environmental sounds, animal voices, music signals, and so on. The noise signals in this corpus were used both for the training set for the SE model and the test sets for ASR with and without SE. A validation set was also made from other data that were not included in either the training or test sets.

The training data for the ASR model were clean speech and reverberated speech convoluted by real impulse responses measured in our laboratory. This augmentation of reverberated speech (reverbaug) made the ASR model more robust against reverberation and Gaussian-like noise signals. We used this noise-robust ASR model for our evaluation.

Table 1: Character error rate (CER: %) for each condition using reverbaug ASR model. \( N \) represents number of samples.

<table>
<thead>
<tr>
<th>Evidence model</th>
<th>Evidence model</th>
<th>( N )</th>
<th>Expectation</th>
<th>eval1</th>
<th>eval2</th>
<th>eval3</th>
<th>CTC, Atten.</th>
<th>( \leq 5% )</th>
<th>( 0 % \leq ) ( 5% )</th>
<th>( 5 % \leq 10 % )</th>
<th>( 10 % \leq 15 % )</th>
<th>( &gt; 15 % )</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>No proc. (with clean model)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>53.7</td>
<td>34.4</td>
<td>17.0</td>
<td>9.7</td>
<td>54.8</td>
<td>35.0</td>
<td>16.0</td>
<td>8.0</td>
</tr>
<tr>
<td>No proc.</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>52.4</td>
<td>29.9</td>
<td>15.3</td>
<td>9.4</td>
<td>53.6</td>
<td>30.6</td>
<td>13.2</td>
<td>7.4</td>
</tr>
<tr>
<td>Baseline (only SE)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>40.0</td>
<td>20.5</td>
<td>11.6</td>
<td>8.2</td>
<td>38.2</td>
<td>18.4</td>
<td>9.2</td>
<td>6.2</td>
</tr>
<tr>
<td>Frame-wise model</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>43.8</td>
<td>22.5</td>
<td>10.6</td>
<td>8.0</td>
<td>43.0</td>
<td>22.9</td>
<td>9.9</td>
<td>6.5</td>
</tr>
<tr>
<td>Proposed (Utterance-wise model)</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>35.7</td>
<td>18.3</td>
<td>10.8</td>
<td>7.9</td>
<td>35.5</td>
<td>16.8</td>
<td>8.7</td>
<td>5.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Evidence model</th>
<th>Evidence model</th>
<th>( N )</th>
<th>Expectation</th>
<th>eval1</th>
<th>eval2</th>
<th>eval3</th>
<th>CTC, Atten.</th>
<th>( \leq 5% )</th>
<th>( 0 % \leq ) ( 5% )</th>
<th>( 5 % \leq 10 % )</th>
<th>( 10 % \leq 15 % )</th>
<th>( &gt; 15 % )</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform: ( \mathcal{E}_{\hat{l}} ) in Eq. (9)</td>
<td>128 prob</td>
<td>128 enc</td>
<td>35.7</td>
<td>18.3</td>
<td>10.8</td>
<td>7.9</td>
<td>35.5</td>
<td>16.8</td>
<td>8.7</td>
<td>5.9</td>
<td>50.0</td>
<td>29.5</td>
<td>18.3</td>
</tr>
<tr>
<td>Delta+Uniform</td>
<td>16 enc</td>
<td>35.1</td>
<td>18.4</td>
<td>10.9</td>
<td>7.9</td>
<td>34.7</td>
<td>16.7</td>
<td>8.7</td>
<td>5.9</td>
<td>48.8</td>
<td>29.4</td>
<td>18.2</td>
<td>13.9</td>
</tr>
<tr>
<td>Gauss+Uniform</td>
<td>128 prob</td>
<td>128 enc</td>
<td>35.5</td>
<td>18.4</td>
<td>10.9</td>
<td>7.9</td>
<td>34.8</td>
<td>16.5</td>
<td>8.7</td>
<td>5.9</td>
<td>49.2</td>
<td>29.3</td>
<td>18.2</td>
</tr>
<tr>
<td>Gaussian</td>
<td>16 enc</td>
<td>35.6</td>
<td>18.4</td>
<td>10.9</td>
<td>7.9</td>
<td>34.7</td>
<td>16.5</td>
<td>8.7</td>
<td>5.9</td>
<td>49.1</td>
<td>29.4</td>
<td>18.2</td>
<td>13.9</td>
</tr>
</tbody>
</table>

The training data for the SE model were a mixture of clean speech and non-speech signals. Each speech signal in the CSJ training set was added from a certain non-speech signal randomly selected from the PSE data. The mixture SNRs were also randomly set from \(-5, 0, 5, 10, 15\) dB.

The test set data were also a mixture of clean speech and non-speech signals with different SNRs to investigate the robustness of our MD-ASR. The clean speech signals were from the CSJ test set, and non-speech signals were also PSE data that were not used for the training set of the SE model. The SNRs were \(-5, 0, 5, 10, 15\) dB.

ii) Configurations of DNNs and MD-ASR

The ASR model was ESPNet [2] with a transformer encoder-decoder acoustic model because it performed best for noisy data. The parameters for STFT were a 512-point Hanning window and 128-point shift. The 80-dimensional mel-filterbank was used for speech features, and the global mean-and-variance normalization was applied. Speech-rate perturbation and `specaug` techniques were also applied to make the model more robust. Please see the ESPNet CSJ recipe and configuration files of ESPNet for details on the networks, RNN language model, and hyper-parameters.

The SE model was built from scratch by using the PyTorch library [1]. The STFT parameters were the same as those of ESPNet. Independent SNRs with same architecture were used for the mask \( m \) and the parameters for precision matrices \( \Lambda_{y,t} \) and \( \Lambda_{\alpha,t} \). Each network consisted of 80-dimensional filterbank networks, absolute and power active functions, concatenation of 32 frames before and after the center frame, layer normalization, and three-layer fully-connected networks with a sigmoid function and dropout layer. The dimension of the middle layer was 2048. The last layer after the linear network was a sigmoid for the mask network, and nothing was used for the last layer of the others, i.e., identity. Gradient clipping [34] and Adam [35] were applied. All the parameters were updated for each utterance, i.e., the mini-batch size was 1. The learning rate was \( 1.0 \times 10^{-4} \). The parameters that performed best for the validation set were used within 50 epochs.

The decoding parameters were also the same as the ESPNet CSJ recipe except for the sampling parameters for the expectation calculation. The beam size was 20, and the weights for the CTC, attention, and the language model were 0.3, 0.7 and 0.3, respectively. For the number of samples \( N \), we evaluated 16, 32, 64, and 128. The mixture weight \( \pi \) was set to 0.25.

4.2. Results

Table 1 shows the character error rate (CER) of each method with different test sets and SNR conditions. A smaller CER means a better ASR performance, and the total number of characters over eval(1,2,3) was 115,745. The column “Evidence model” represents the type of evidence model, i.e., Gaussian \( (\mathcal{E}_p) \), Uniform \( (\mathcal{E}_{\hat{l}}) \), Delta+Uniform \( (\mathcal{E}_{\hat{l}}, \bar{\mathcal{E}}_{\hat{l}}) \), and Gauss+Uniform \( (\mathcal{E}_{\hat{l}}, \bar{\mathcal{E}}_{\hat{l}}) \) pdf. The “Expectation” column means which expectation was used in the CTC and attention networks, encoded-vectors (enc) or CTC-probability (prob) as shown in Fig. 2. “No proc.” means the performance of the direct recognition of the noisy observed signals. The average CER with reverbaug model was 1.4-point better than that with the model trained only by clean speech. The method with only SE was a baseline.

We found that the expectation of the encoded vectors performed better than that of CTC probability in both cases of Uniform and Gauss+Uniform pdf. The averaged CERs of “prob” degraded by 0.2 and 0.5 points from that of “enc” with both evidence models. The consistency among the two networks may be important for accurate recognition.

The Gauss+Uniform pdf performed best among the other evidence models by 0.4 points on average, which indicates the usefulness of the uncertainty from the speech enhancement. Since the difference between Uniform and Delta+Uniform was small, the Uniform pdf which uses only enhanced speech and observed signals, played a fundamental role in all of the evidence models. The random sampling from the frame-wise posterior of SE (Gauss: Eq. (12)) did not work and marked the worst CER among all evidence models. This is because the continuity as a sequential feature was lost for these samples due to the model’s assumption: independent among frames.

The number of samples \( N \) did not affect the performances seriously as seen in a CER comparison with \( N = 128 \) and 16. This means that the latent space of our evidence models was small enough or we need to increase the degree of freedom of the latent evidence model to utilize more samples effectively.

5. Conclusion

In this paper, we proposed an empirical sampling from the latent space of an utterance-wise evidence model. We selected or generated a limited number of samples from the latent space while keeping sequential continuity. Experiments showed that the character error rate of the enhanced speech was further improved by 2.5 points on average with our MD-ASR.

Future work includes a more flexible utterance-wise posterior pdf of SE and an efficient method for sampling from it. The sequential variational autoencoder approach may solve both of these problems because we can build a latent probabilistic model into DNNs. We will also investigate the influence of the acoustic model trained with noisy speech.
6. References


