Incremental Binarization on Recurrent Neural Networks For Single-Channel Source Separation

SAIGE
INDIANA UNIVERSITY
SIGNALS & ARTIFICIAL INTELLIGENCE GROUP IN ENGINEERING
http://saige.sice.indiana.edu

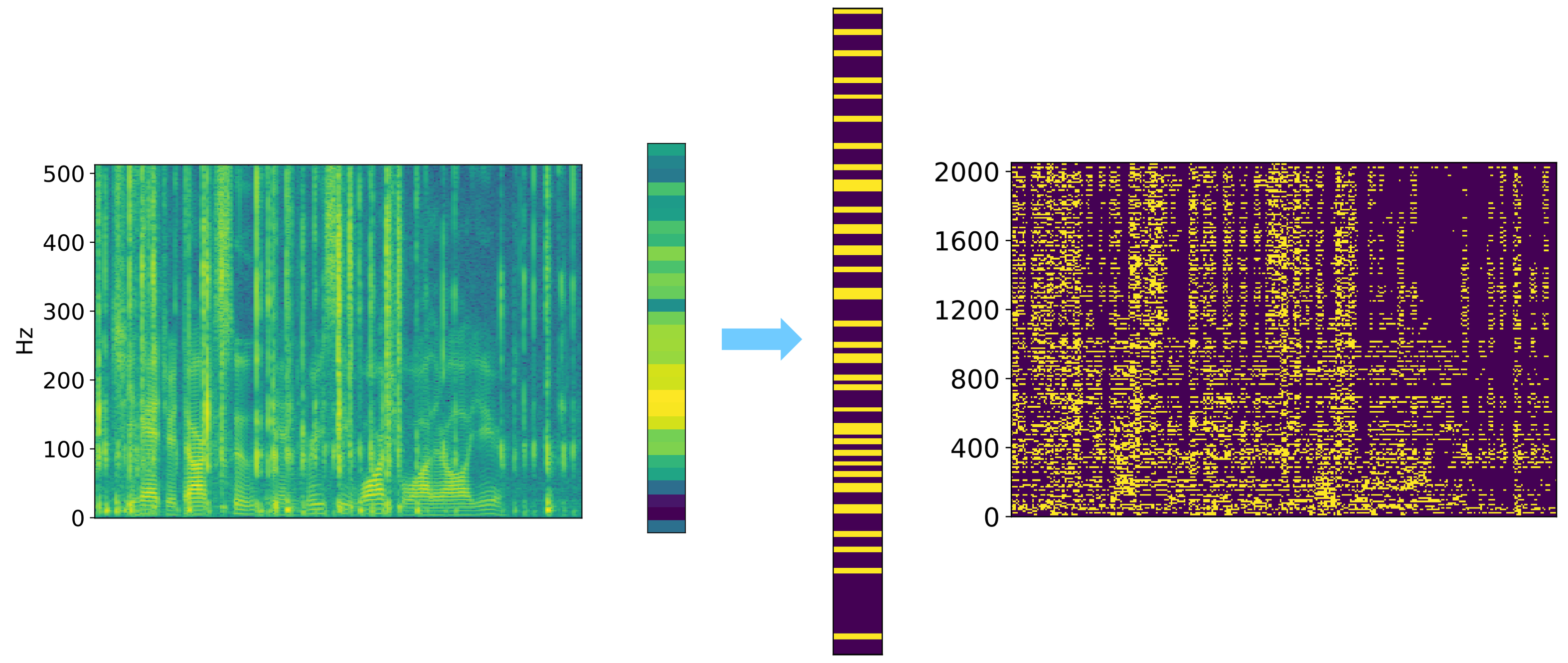
Sunwoo Kim, Mrinmoy Maity, Minje Kim kimsunw@indiana.edu, mmaity@iu.edu , minje@Indiana.edu
Indiana University
Department of Intelligent Systems Engineering

INTRODUCTION

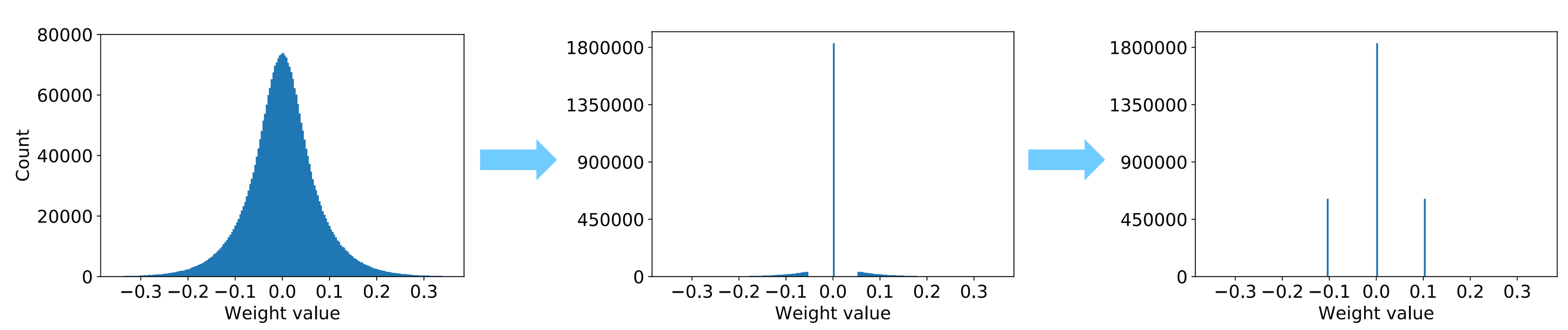
We propose a **Bitwise Gated Recurrent Unit (BGRU)** network for the single-channel source separation task that mitigates the computation required by Recurrent Neural Networks. By re-defining the originally real-valued inputs and outputs, pretrained weights, and operations in a bitwise fashion, we reduce the computational and spatial complexity of the GRU network. To address the heavy quantization loss from the transformation, we take an incremental approach to binarization.

QUANTIZATION

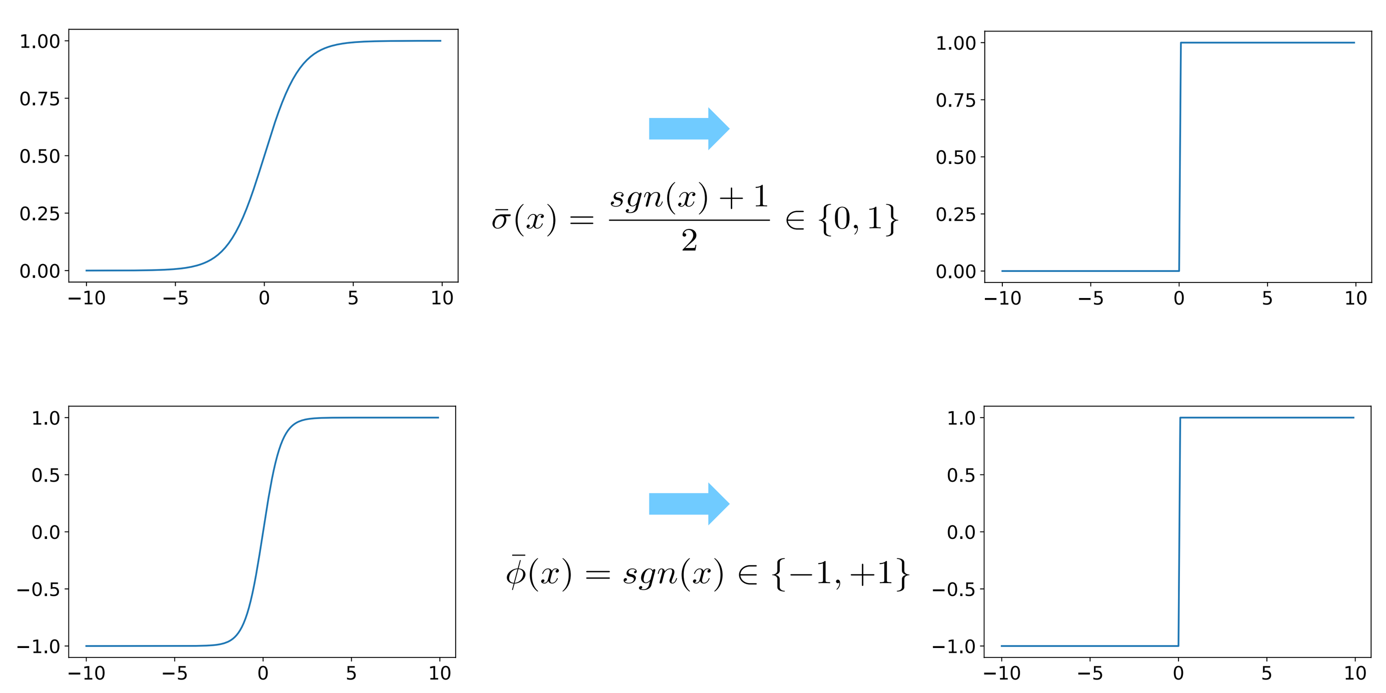
- Input STFT magnitude bins are quantized into 4 binary bits using Quantization-and-Dispersion



- Pretrained weights are transformed and scaled with a relaxed quantization on a boundary determined by a specified sparsity level



- Binary versions of the logistic and hyperbolic tangent activations

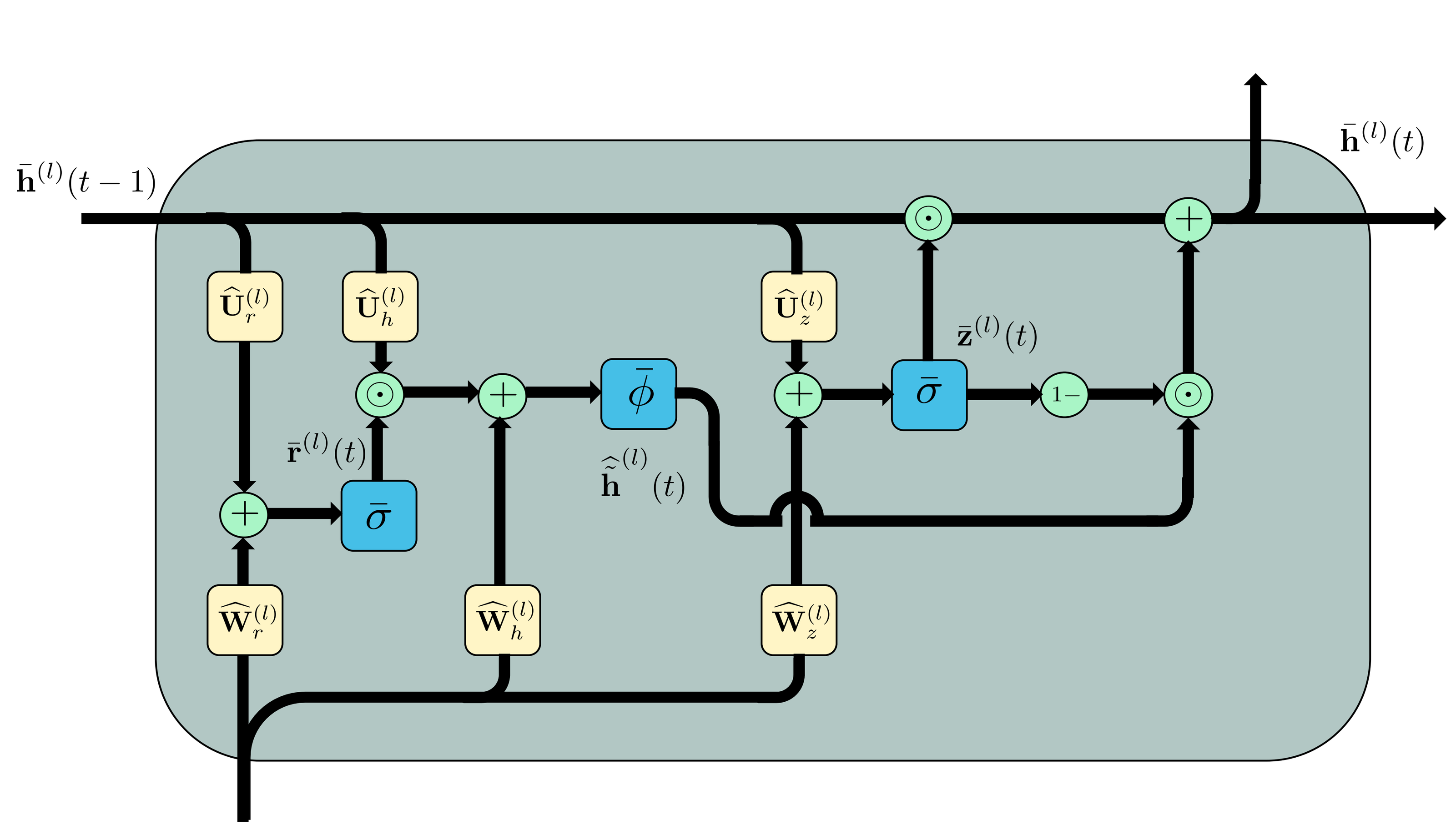


$$\bar{\sigma}(x) = \frac{sgn(x)+1}{2} \in \{0,1\}$$

$$\bar{\phi}(x) = sgn(x) \in \{-1,+1\}$$

PROPOSED MODEL

- BGRU Cell



- **Feedforward**: real-valued weights are incrementally binarized by scaled sparsity and Bernoulli masks.
  - Example of candidate state:

$$\widehat{\mathbf{W}}_h^{(l)} = (\bar{\phi}(\mathbf{W}_h^{(l)}) \odot \mathbf{B}) \odot \mathbf{C} + \phi(\mathbf{W}_h^{(l)}) \odot (1-\mathbf{C})$$

$$\widehat{\mathbf{U}}_h^{(l)} = (\bar{\phi}(\mathbf{U}_h^{(l)}) \odot \mathbf{B}) \odot \mathbf{C} + \phi(\mathbf{U}_h^{(l)}) \odot (1-\mathbf{C})$$

$$\mathbf{V} = \widehat{\mathbf{W}}_h^{(l)} \mathbf{x}^{(l-1)}(t) + \widehat{\mathbf{U}}_h^{(l)} \big(\bar{\mathbf{r}}^{(l)}(t) \odot \bar{\mathbf{h}}^{(l)}(t-1)\big)$$

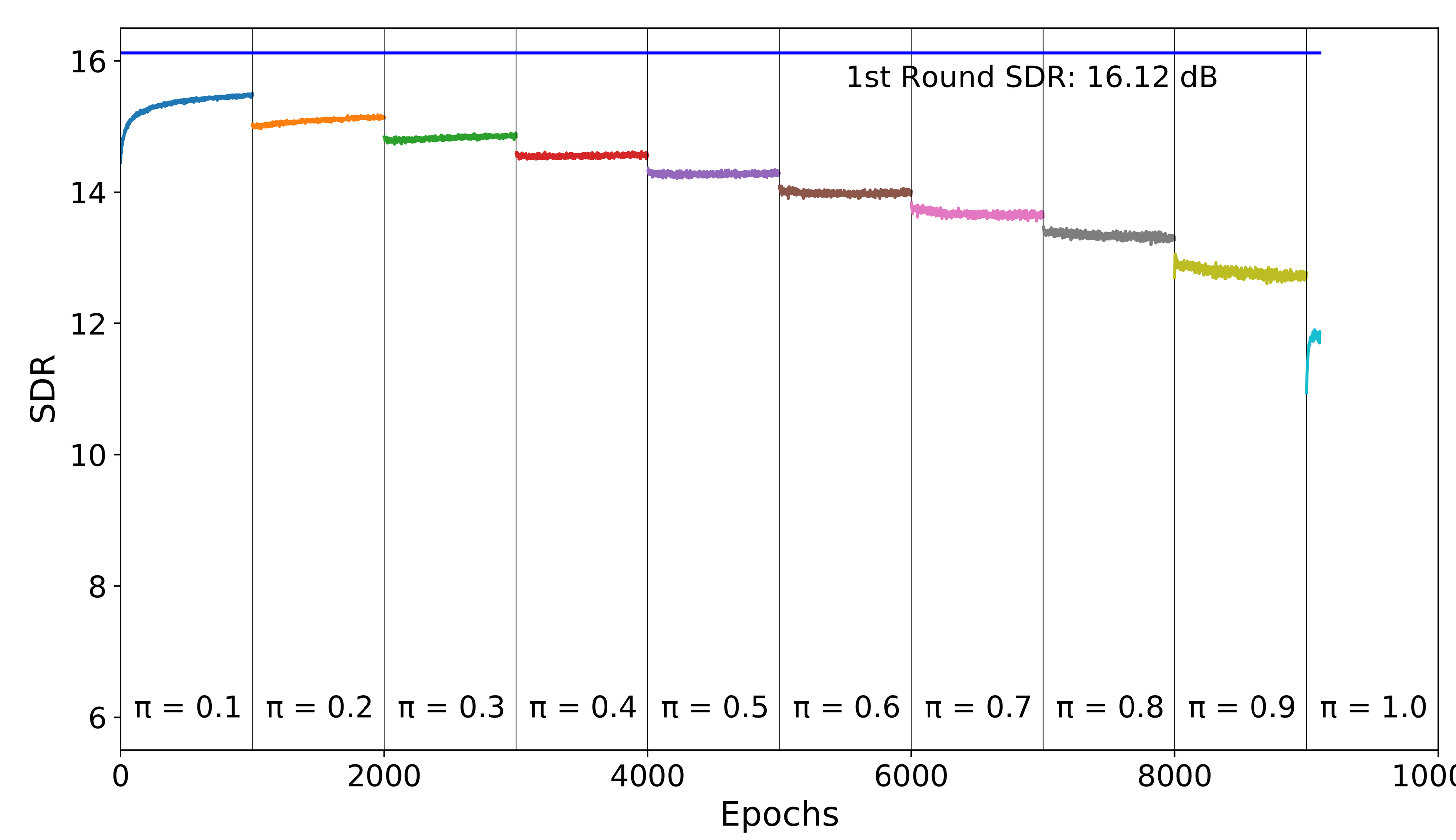
$$\widehat{\tilde{\mathbf{h}}}^{(l)}(t) = \bar{\phi}(\mathbf{V}) \odot \mathbf{C} + \phi(\mathbf{V}) \odot (1-\mathbf{C})$$

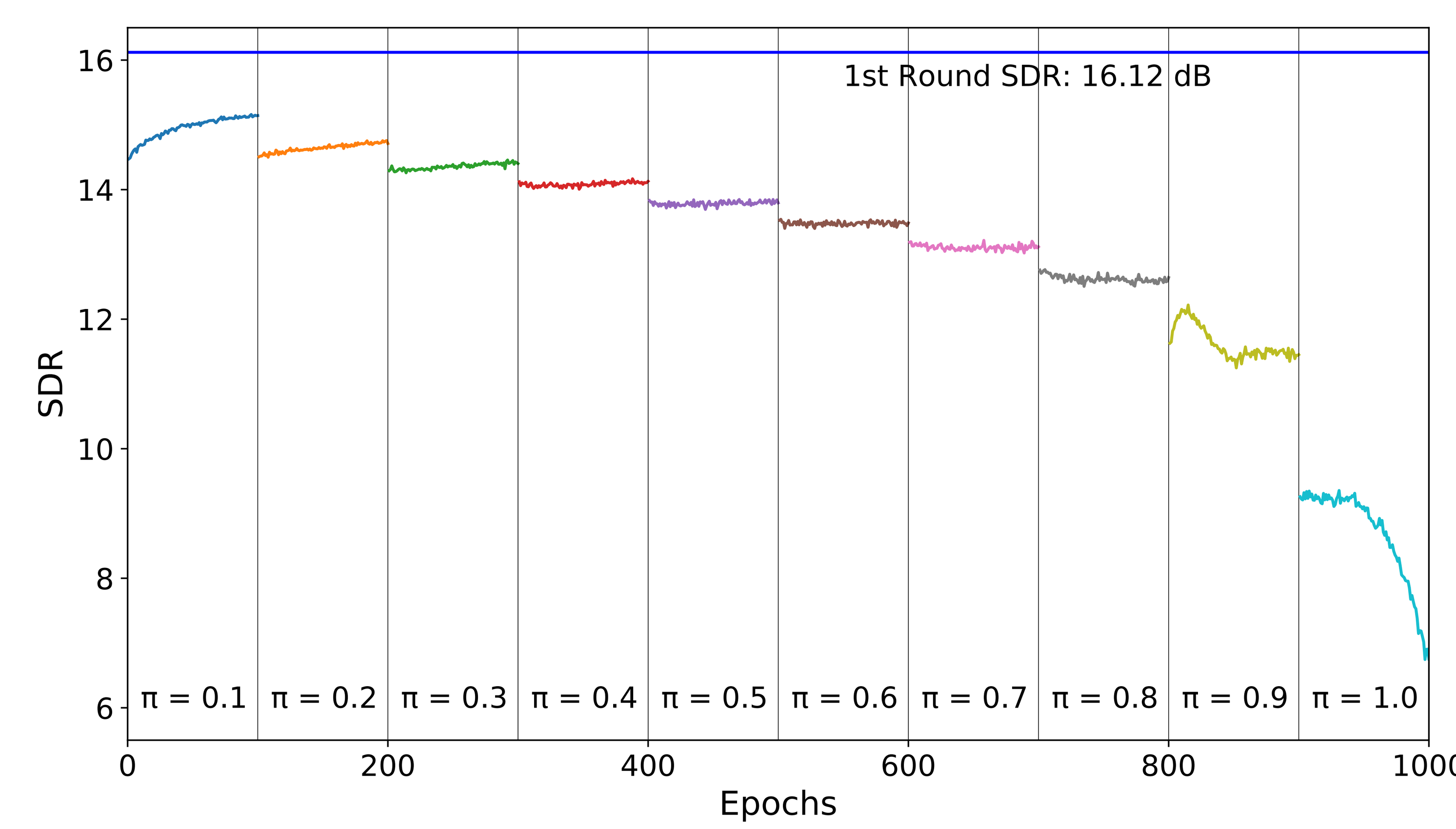
- **Backpropagation**: Derivatives of non-differentiable activation functions are overwritten with that of relaxed counterparts
  - Example of candidate state:

$$\nabla \mathbf{W}_h^{(l)} = \nabla \mathbf{W}_h^{(l)} \odot (\mathbf{B} \odot \mathbf{C} + (1-\mathbf{C}))$$

$$\nabla \mathbf{U}_h^{(l)} = \nabla \mathbf{U}_h^{(l)} \odot (\mathbf{B} \odot \mathbf{C} + (1-\mathbf{C}))$$

EXPERIMENTAL RESULTS





Systems		Topology	SDR	STOI
FCN with original input		1024×2	10.17	0.7880
		2048×2	10.57	0.8060
FCN with binary input		1024×2	9.80	0.7790
		2048×2	10.11	0.7946
BNN		1024×2	9.35	0.7819
		2048×2	9.82	0.7861
GRU with binary input		1024×1	16.12	0.9459
BGRU	π=0.1	1024×1	15.50	0.9393
	π=0.2		15.17	0.9361
	π=0.3		14.90	0.9324
	π=0.4		14.58	0.9292
	π=0.5		14.32	0.9252
	π=0.6		14.02	0.9217
	π=0.7		13.66	0.9174
	π=0.8		13.30	0.9104
	π=0.9		12.70	0.9019
	π=1.0		11.76	0.8740

CONCLUSION

- Training is done in two rounds, first in a weight compressed network then in an incrementally bitwise version with the same topology
- Due to the sensitivity in training the BGRU network, the bitwise feedforward pass is performed gently using two types of masks that determine the level of sparsity and rate of binarization.

REFERENCES

- M. Kim and P. Smaragdis, “Bitwise neural networks for effi-cient single-channel source separation,” in2018 IEEE Interna-tional Conference on Acoustics, Speech and Signal Processing(ICASSP). IEEE, 2018, pp. 701–705.
- M. Courbariaux, Y. Bengio, and J. P. David, “BinaryConnect: Training deep neural networks with binary weights dur-ing propagations,” inAdvances in neural information process-ing systems, 2015, pp. 3123–3131.
- I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, andY. Bengio, “Binarized neural networks,” inAdvances in neuralinformation processing systems, 2016, pp. 4107–4115.
- Joachim Ott, Zhouhan Lin, Ying Zhang, Shih-Chii Liu, and Yoshua Bengio, “Recurrent neural networks with limited numerical precision,” arXiv preprint arXiv:1608.06902, 2016.

This project was supported by Intel Corporation