Limiting Numerical Precision of Neural Networks to Achieve Real-time Voice Activity Detection

Jong Hwan Ko[*], Josh Fromm[†],
Matthai Philipose[‡], Ivan Tashev[‡], Shuayb Zarar[‡]

[*]Georgia Tech, [†]Univ. of Washington, [‡]Microsoft AI and Research

April 18, 2018

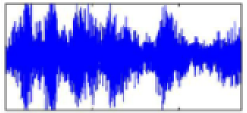
Outline

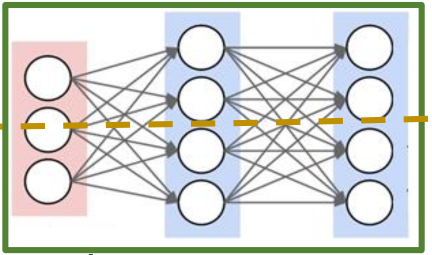
- Introduction
- Bit Precision Control
- Experimental Framework & Speech Dataset
- Performance Evaluation
  - Evaluation of Model-based/DNN-based Approaches
  - Effect of Seen/Unseen Noise
  - Effect of Bit Precision Control
  - Network Optimization
- Conclusion

Research Objectives

Neural network

Noisy speech





Voice activity detection

…0110101…

Real-time applications

→ Processing **delay** ↓

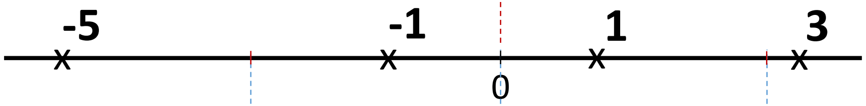
Design of **resource-efficient** neural networks using **bit precision control** of weights/neurons

Bit Precision Control
How it works

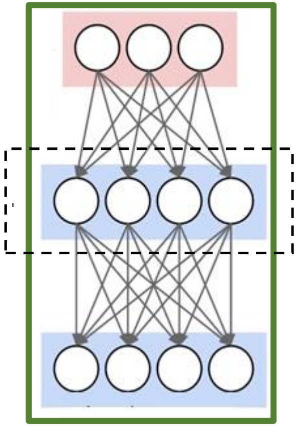
Bit assignment: **0** | **1**

Avg. distance from 0 (d_1)= 2.5

Approx. values: $\mu_0 = -d_1 = -2.5$ | $\mu_1 = d_1 = 2.5$



Original 32-bit values

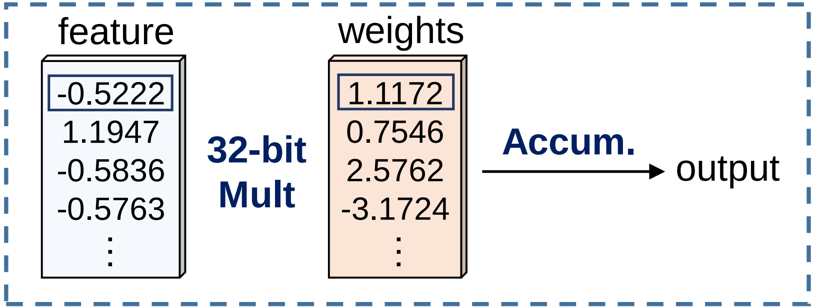


Avg. distance from μ_0, μ_1 (d_2)= 1.5

Bit assignment	**00**	**01**	**10**	**11**
Approx. values	μ_{00} $= -\mu_0 - d_2$ $= -2.5-1.5$ $=-4$	μ_{01} $= -\mu_0 + d_2$ $=-2.5+1.5$ $=-1$	μ_{10} $= \mu_1 - d_2$ $=2.5-1.5$ $=1$	μ_{11} $= \mu_0 + d_2$ $=2.5+1.5$ $=4$

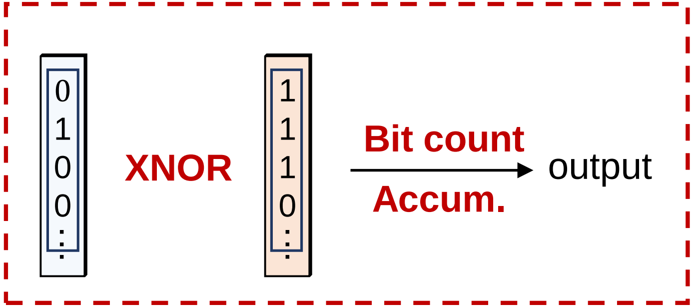
Bit Precision Control
Why it is beneficial

32-bit network



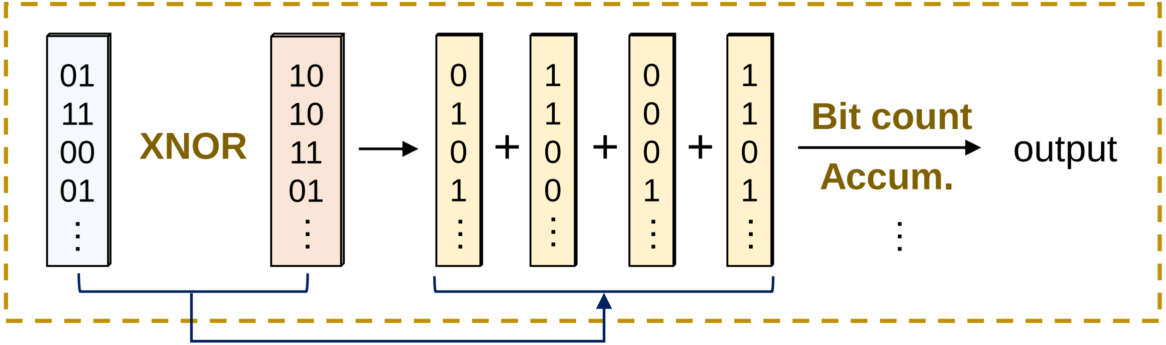
2 operations per **1 element**

1-bit network



3 operations per **64 elements**
(43x speedup)

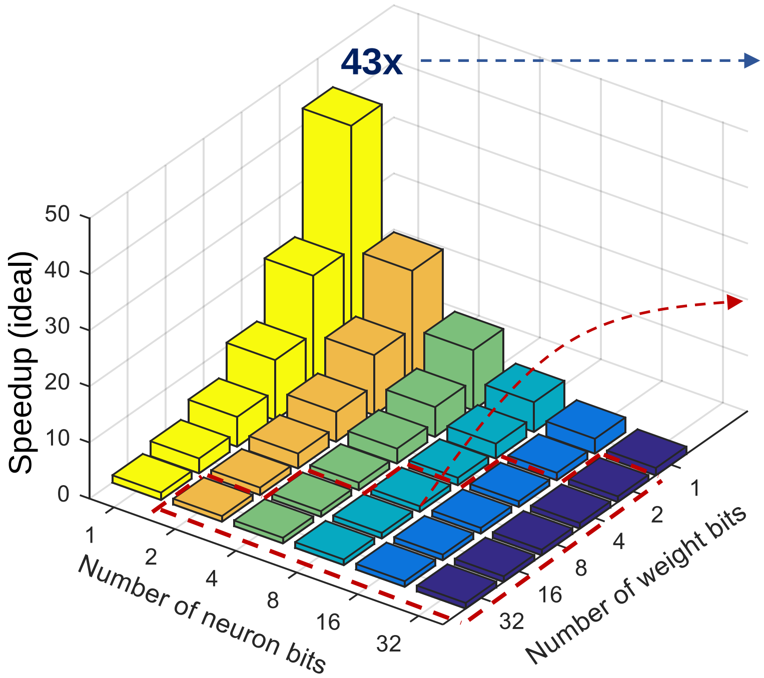
2-bit network



4 combinations → **4x3 operations** per **64 elements (10.7x speedup)**

Bit Precision Control
How it is beneficial

- Ideal inference speedup
  = Max(1, 43 / (# weight bits x # neuron bits))



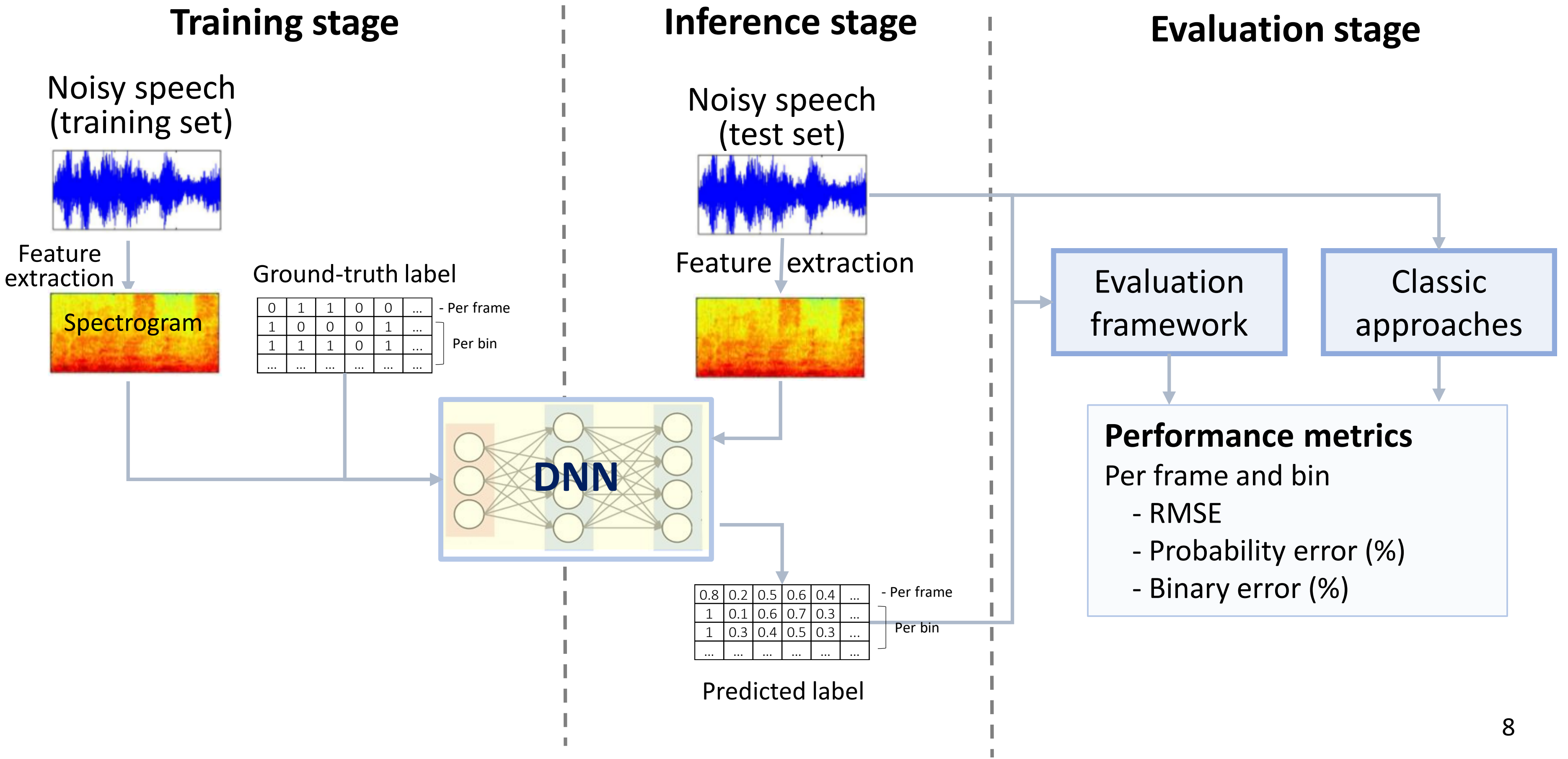
Actual measurement: ~30x speedup

(# weight bits x # neuron bits) **≥ 43**
→ **No advantage** from precision control
→ Use original multiplication than XNOR
(**1x** improvement)

Outline

- Introduction
- Bit Precision Control
- **Experimental Framework & Speech Dataset**
- Performance Evaluation
  - Evaluation of Model-based/DNN-based Approaches
  - Effect of Seen/Unseen Noise
  - Effect of Bit Precision Control
  - Network Optimization
- Conclusion

Experimental Framework



Speech Dataset

Clean speech

- Voice queries to Cortana
- 10 utterances each (Duration: 0’55” – 1’30”)

\+ convolution with randomly selected room impulse response +

Noise

- Subset of the MS noise collection
- 377 files with 25 types

↓

- Training set: 750 files
- Validation set: 150 files
- Test set: 150 files

Unseen noise

- NOISEX-92 noise corpus + MS noise collection
- 32 files with 32 types

↓

- Test set with unseen noises: 150 files

Outline

- Introduction
- Bit Precision Control
- Experimental Framework & Speech Dataset
- **Performance Evaluation**
  - **Evaluation of Model-based/DNN-based Approaches**
  - **Effect of Seen/Unseen Noise**
  - **Effect of Bit Precision Control**
  - **Network Optimization**
- Conclusion

Existing Model-Based Approaches

- Classic VAD (MATLAB)
  - Assuming Gaussian distribution of speech and noise signal*

$$\Lambda_k = \frac{p(X_k \mid H_1)}{p(X_k \mid H_0)} = \frac{\frac{2X_k}{\left(\lambda_S(k)+\lambda_N(k)\right)} \exp\left(-\frac{|X_k|^2}{\lambda_S(k)+\lambda_N(k)}\right)}{\frac{2X_k}{\lambda_N(k)} \exp\left(-\frac{|X_k|^2}{\lambda_N(k)}\right)} = \frac{1}{1+\xi_k} \exp\left(\frac{\gamma_k \xi_k}{1+\xi_k}\right)$$

* Sohn and Sung, 1998 and 1999
† Ephraim and Malah, 1984
‡ Tashev et al, 2010

  - Prior SNR estimation†, hangover scheme*, combining the likelihoods per bin‡,…

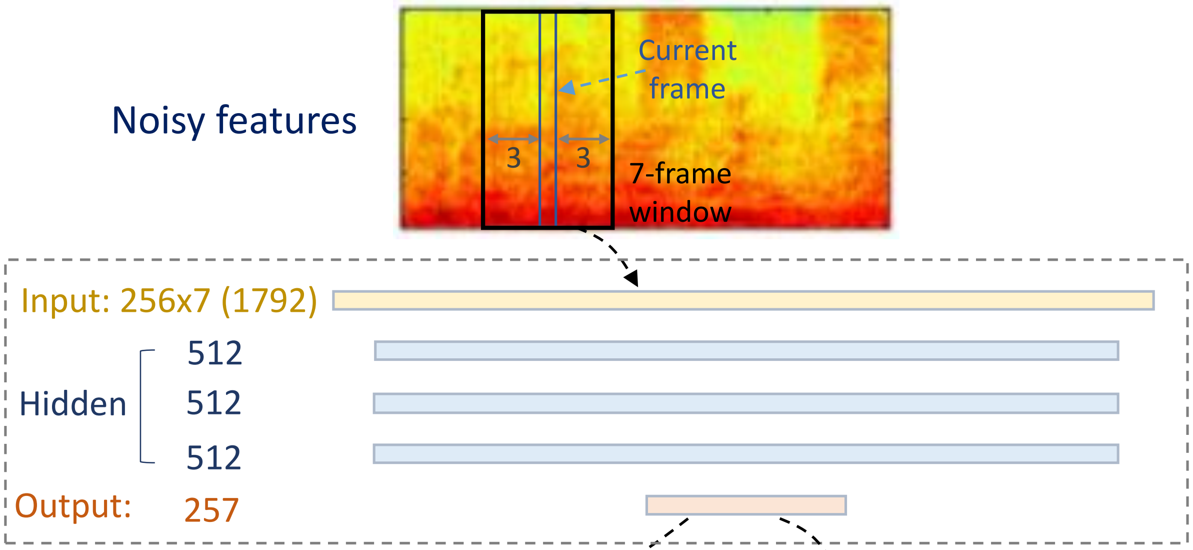
- Google WebRTC VAD (python)
  - WebRTC
    - Open source project for web browsers with real-time capabilities
  - WebRTC VAD
    - Reportedly one of the best available, being fast
    - The probability of speech is calculated by the Gaussian mixture model

Existing Model-Based Approaches

Metric		Clean speech		Noisy speech	
		Classic	WebRTC	Classic	WebRTC
Per frame error	RMSE	0.335	**0.255**	0.411	**0.408**
	Probability (%)	12.90	-	24.24	-
	Binary (%)	12.70	**6.70**	24.90	**20.46**

- Works well with **clean speech**
- Speech with **noise** → VAD **error** ↑

DNN Model



0	1	1	0	0	1	1	0	…
1	0	0	0	1	0	0	0	…
1	1	1	0	1	1	1	0	…
…	…	…	…	…	…	…	…	…

Ground-truth Labels

0	1	1	0	0	1	1	0	…
1	0	0	0	1	0	0	0	…
1	1	1	0	1	1	1	0	…
…	…	…	…	…	…	…	…	…

Predicted Labels

- Network parameters
  - Based on prior work †
  - Loss function: squared error between spectrogram features
  - Activation: tanh
- Training parameters
  - Batch size: 400
  - 100 epochs

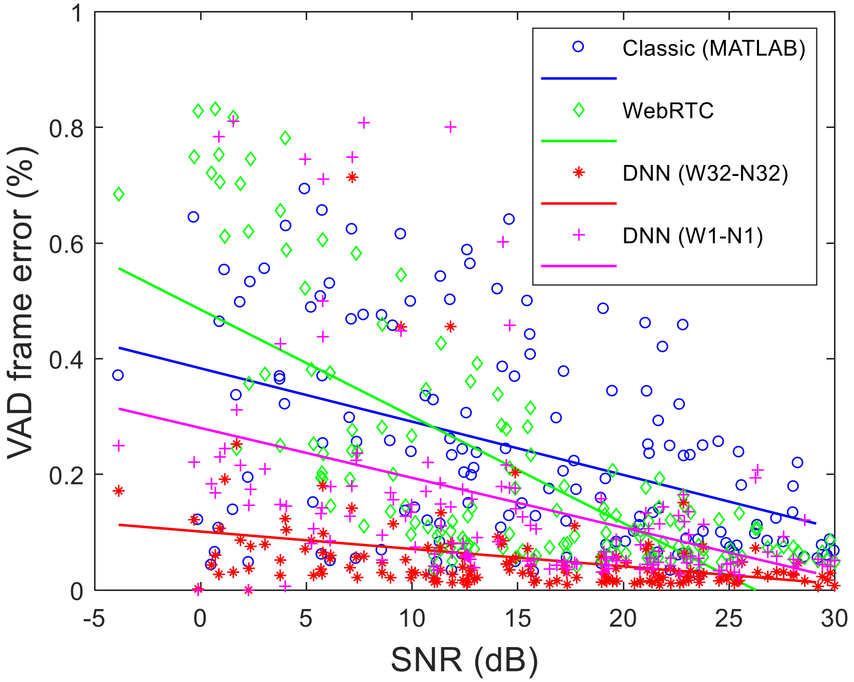
† I. Tashev and S. Mirsamadi, ITA 2016

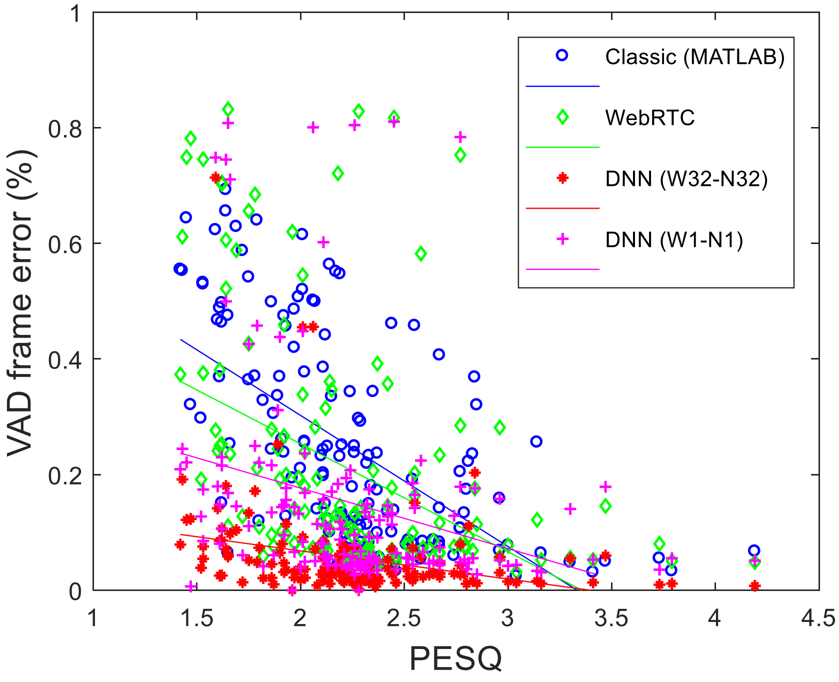
Performance Comparison with Noisy Speech

Model		Classic	WebRTC	DNN	
				W32_W32	W1_N1
Per frame error	RMSE	0.411	0.408	**0.268**	0.389
	Probability (%)	24.24	-	**5.96**	21.63
	Binary (%)	24.90	20.46	**5.55**	14.95

DNNs show **lower error than model-based methods**
(even with **1-bit** network)

Speech Quality vs VAD Error





- Speech **quality** ↓ → VAD **error** ↑
- **DNNs** generally perform **better than model-based methods**

Evaluation Example

Detected voice

Ground-truth

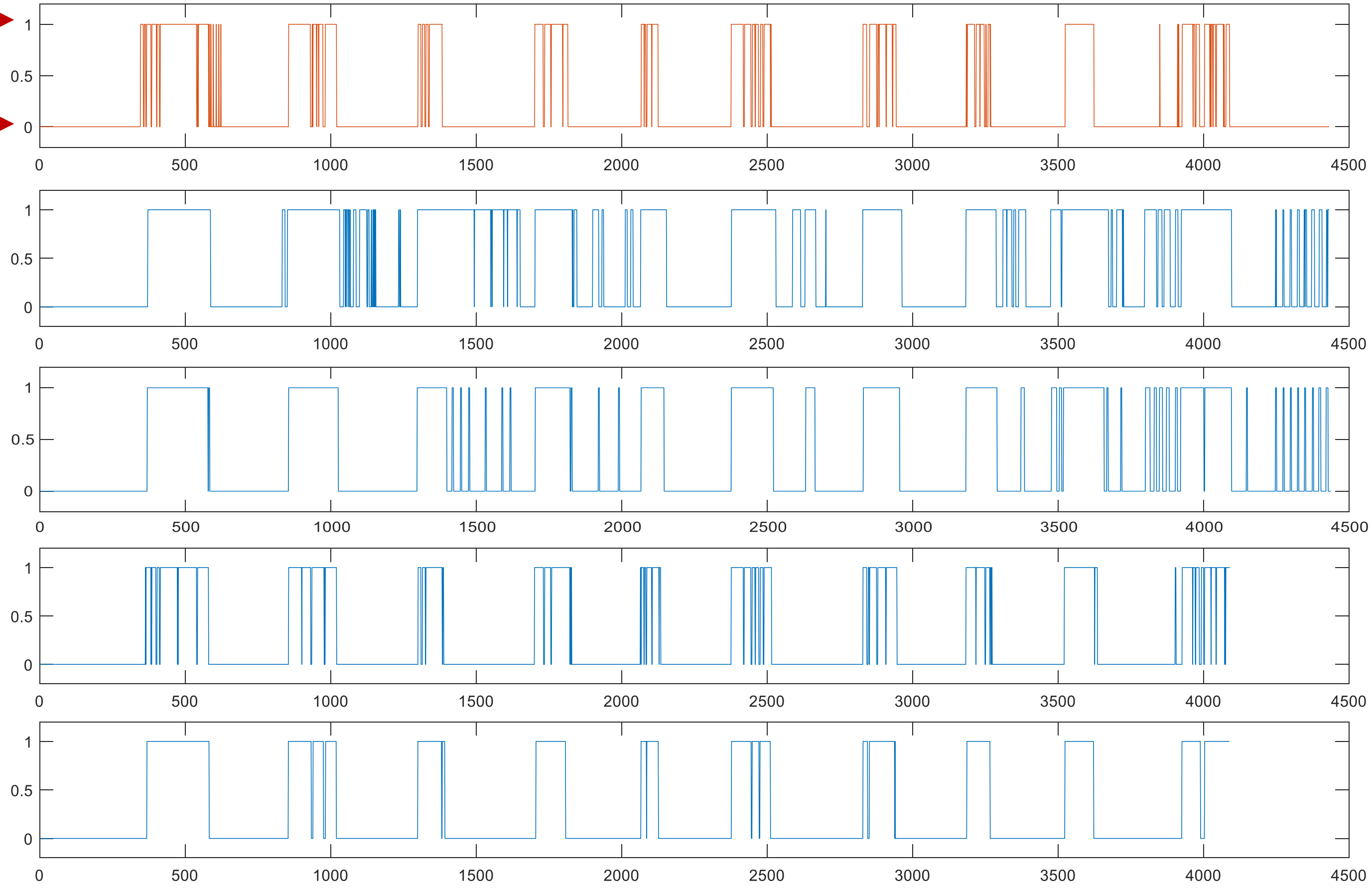
No voice

Classic
(23.4%)

WebRTC
(12.1%)

DNN-W32_N32
(4.1%)

DNN-W1_N1
(14.38%)



Effect of Seen/Unseen Noise

Metric	Classic		WebRTC		DNN			
					W32-N32		W1-N1	
	Seen noise	Unseen noise	Seen noise	Unseen noise	Seen noise	Unseen noise	Seen noise	Unseen noise
RMSE	0.411	**0.3434**	0.4079	**0.3888**	0.268	**0.228**	0.389	**0.312**
Probability (%)	24.24	**17.89**	-	-	**5.96**	15.32	**21.63**	24.51
Binary (%)	24.90	**18.08**	**20.46**	20.88	**5.55**	8.20	**14.95**	17.76

When **test set** has **different noise profile** than **training set**

→ **Model-based** methods perform **similar,**

DNN-based methods perform **worse**

Effect of Bit Precision Control

VAD frame error (%)

Model	N32	N8	N4	N2	N1
W32	8.20				
W8		8.65	8.75	9.45	14.70
W4		8.71	8.57	9.97	14.83
W2		10.02	10.34	9.93	14.84
W1		11.35	12.18	11.34	17.76

Normalized **speedup /**
Normalized **VAD frame error**

Model	N32	N8	N4	N2	N1
W32	0				
W8		0	0.1324	0.3005	0.1506
W4		0.1428	1.0151	0.553	0.333
W2		0.2064	0.4574	1.2762	0.6856
W1		0.3107	0.5547	1.2807	1

Precision of **weights bits** has **less impact** on performance than **neuron bits**

Optimal pair for latency and error
(for this network & dataset)
= **<2-bit neurons, 1-bit weights>**

Processing Delay Measurement

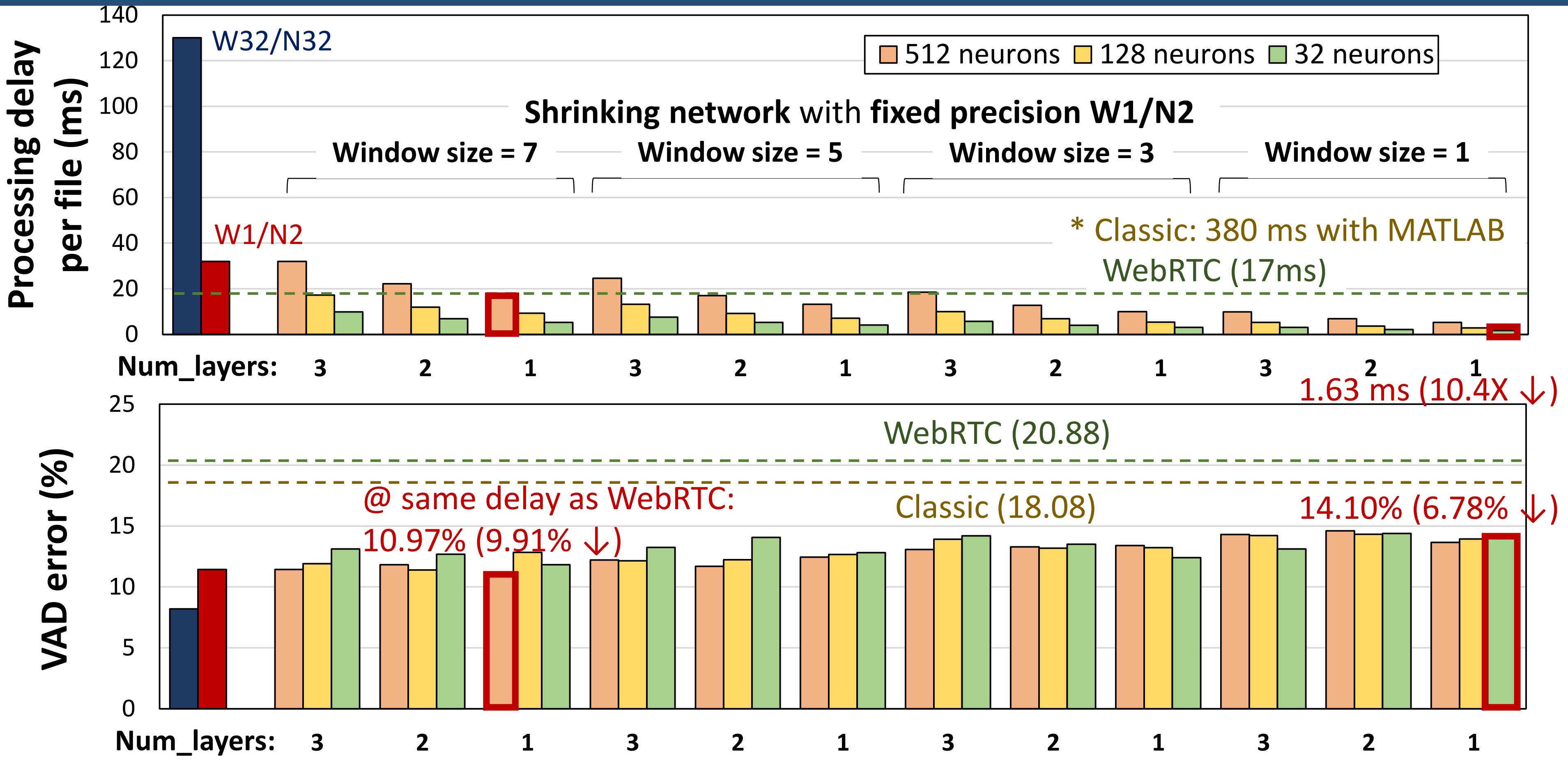
	Classic	WebRTC	DNN	
			W32/N32	W1/N2
Platform	MATLAB	Python		
Processing delay per file (ms)	380	17	138	30.7 (4.5x speedup)

* With 8-core 3.5 GHz CPU machine

Further reduce delay by optimizing the network

- Number of layers
- Number of neurons
- Window size

Network Optimization



Conclusion

- Designed efficient neural networks for real-time VAD using bit precision control

- VAD performance
  - **Baseline** DNN: much **lower error (5.55%)** than classic approaches (20%~)
  - Optimization of bit precision (W1_N2) and network size
    → **10x lower delay/6.78% lower error** than Google WebRTC VAD
    (half the error at the same delay)

- Future work
  - Application to other tasks
    - Classification tasks - source separation and microphone beam forming
    - Estimation tasks - acoustic echo cancellation


Thank you!!