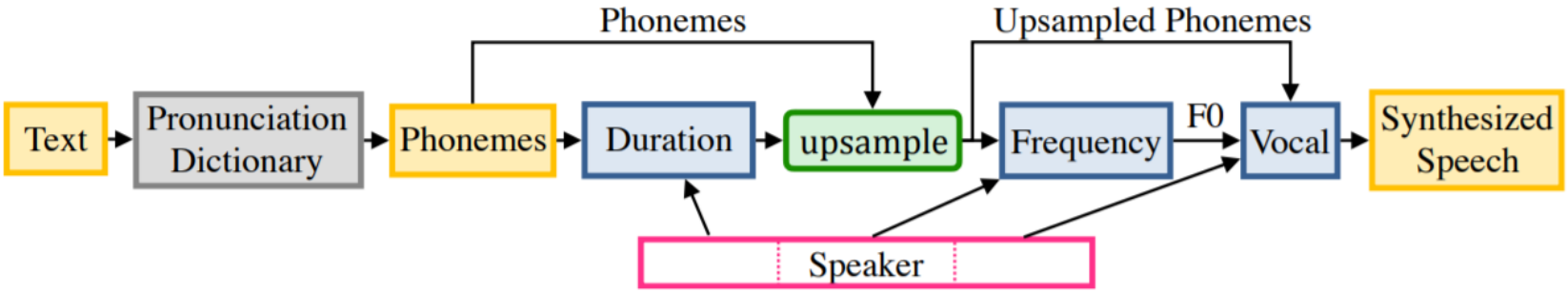
Speech in NIPS 2017/2018

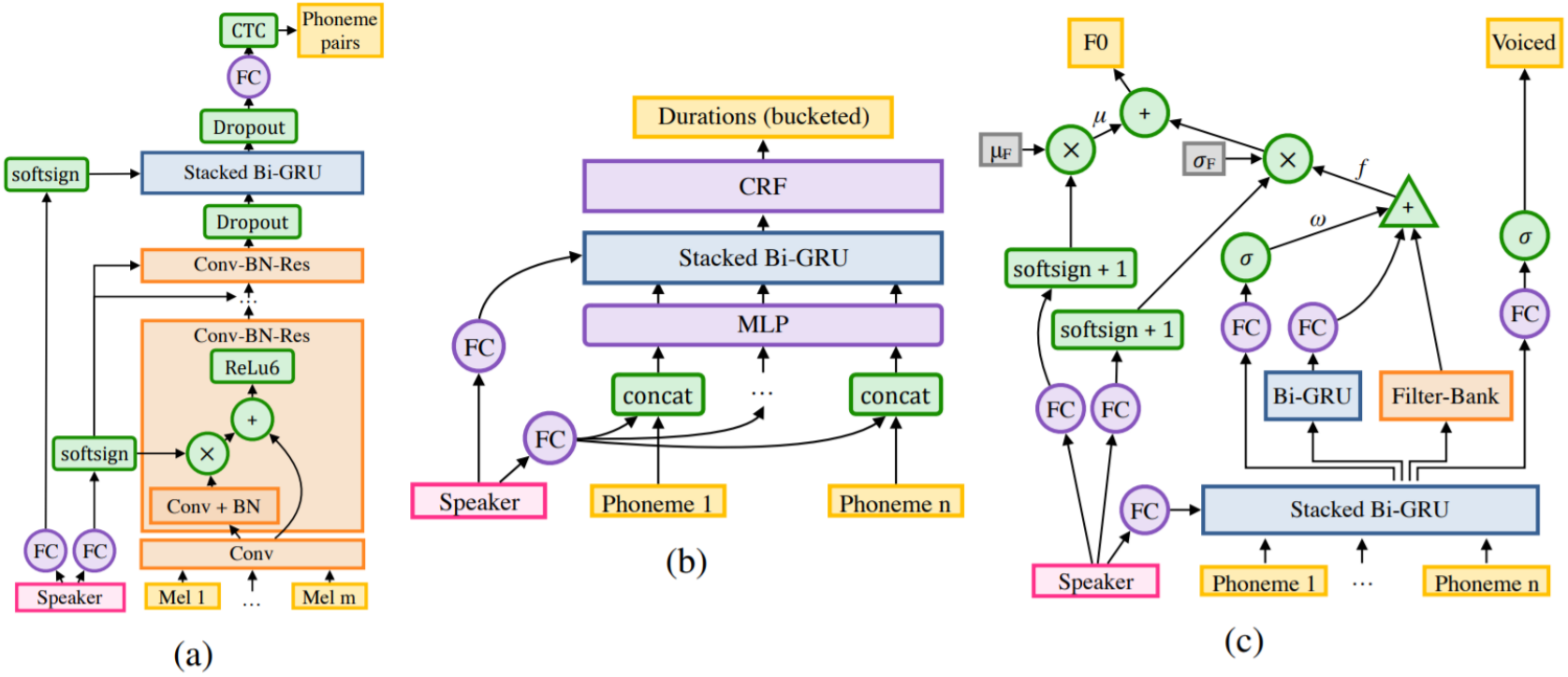
Dong Wang

2019/01/04

Deep Voice 2: Multi-Speaker Neural Text-to-Speech



1. Segmentation model: CTC-based CNN-RNN
2. Duration model: CRF
3. Frequency model: predict silence/phone and F0
4. Vocoder: WaveNet
5. Speaker model: d-vector



Speaker model:

1. VCTK, 44 hours of speech, 108 speakers
2. Baidu Internal, 477 spekers, 30 minutes for each

Model	Samp. Freq.	MOS
Deep Voice 1	16 KHz	2.05 ± 0.24
Deep Voice 2	16 KHz	2.96 ± 0.38
Tacotron (Griffin-Lim)	24 KHz	2.57 ± 0.28
Tacotron (WaveNet)	24 KHz	4.17 ± 0.18

Table 1: Mean Opinion Score (MOS) evaluations with 95% confidence intervals of Deep Voice 1, Deep Voice 2, and Tacotron. Using the crowdMOS toolkit, batches of samples from these models were presented to raters on Mechanical Turk. Since batches contained samples from all models, the experiment naturally induces a comparison between the models.

Dataset	Multi-Speaker Model	Samp. Freq.	MOS	Acc.
VCTK	Deep Voice 2 (20-layer WaveNet)	16 KHz	2.87±0.13	99.9%
VCTK	Deep Voice 2 (40-layer WaveNet)	16 KHz	3.21±0.13	100 %
VCTK	Deep Voice 2 (60-layer WaveNet)	16 KHz	3.42±0.12	99.7%
VCTK	Deep Voice 2 (80-layer WaveNet)	16 KHz	3.53±0.12	99.9%
VCTK	Tacotron (Griffin-Lim)	24 KHz	1.68±0.12	99.4%
VCTK	Tacotron (20-layer WaveNet)	24 KHz	2.51±0.13	60.9%
VCTK	Ground Truth Data	48 KHz	4.65±0.06	99.7%
Audiobooks	Deep Voice 2 (80-layer WaveNet)	16 KHz	2.97±0.17	97.4%
Audiobooks	Tacotron (Griffin-Lim)	24 KHz	1.73±0.22	93.9%
Audiobooks	Tacotron (20-layer WaveNet)	24 KHz	2.11±0.20	66.5%
Audiobooks	Ground Truth Data	44.1 KHz	4.63±0.04	98.8%

Table 2: MOS and classification accuracy for all multi-speaker models. To obtain MOS, we use crowdMOS toolkit as detailed in Table 1. We also present classification accuracies of the speaker discriminative models (see Appendix E for details) on the samples, showing that the synthesized voices are as distinguishable as ground truth audio.

Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis

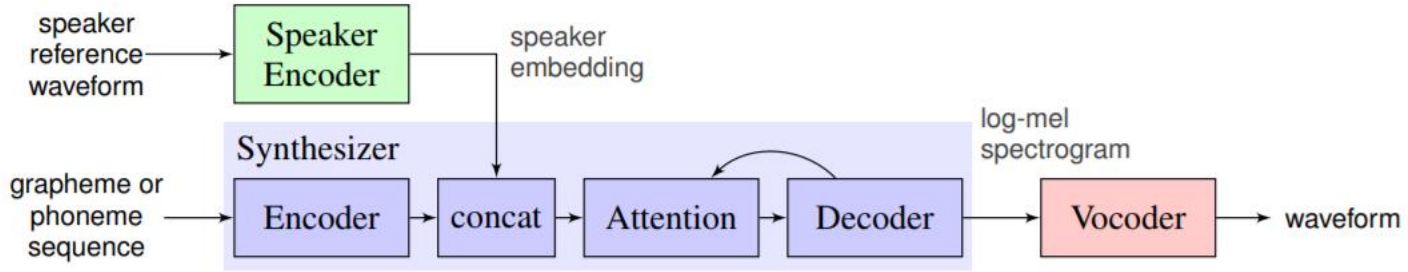


Figure 1: Model overview. Each of the three components are trained independently.

- D-vector speaker encoding
- Tacotron2
- VCTK and LibriSpeech

Table 1: Speech naturalness Mean Opinion Score (MOS) with 95% confidence intervals.

System	VCTK Seen	VCTK Unseen	LibriSpeech Seen	LibriSpeech Unseen
Ground truth	4.43 ± 0.05	4.49 ± 0.05	4.49 ± 0.05	4.42 ± 0.07
Embedding table	4.12 ± 0.06	N/A	3.90 ± 0.06	N/A
Proposed model	4.07 ± 0.06	4.20 ± 0.06	3.89 ± 0.06	4.12 ± 0.05

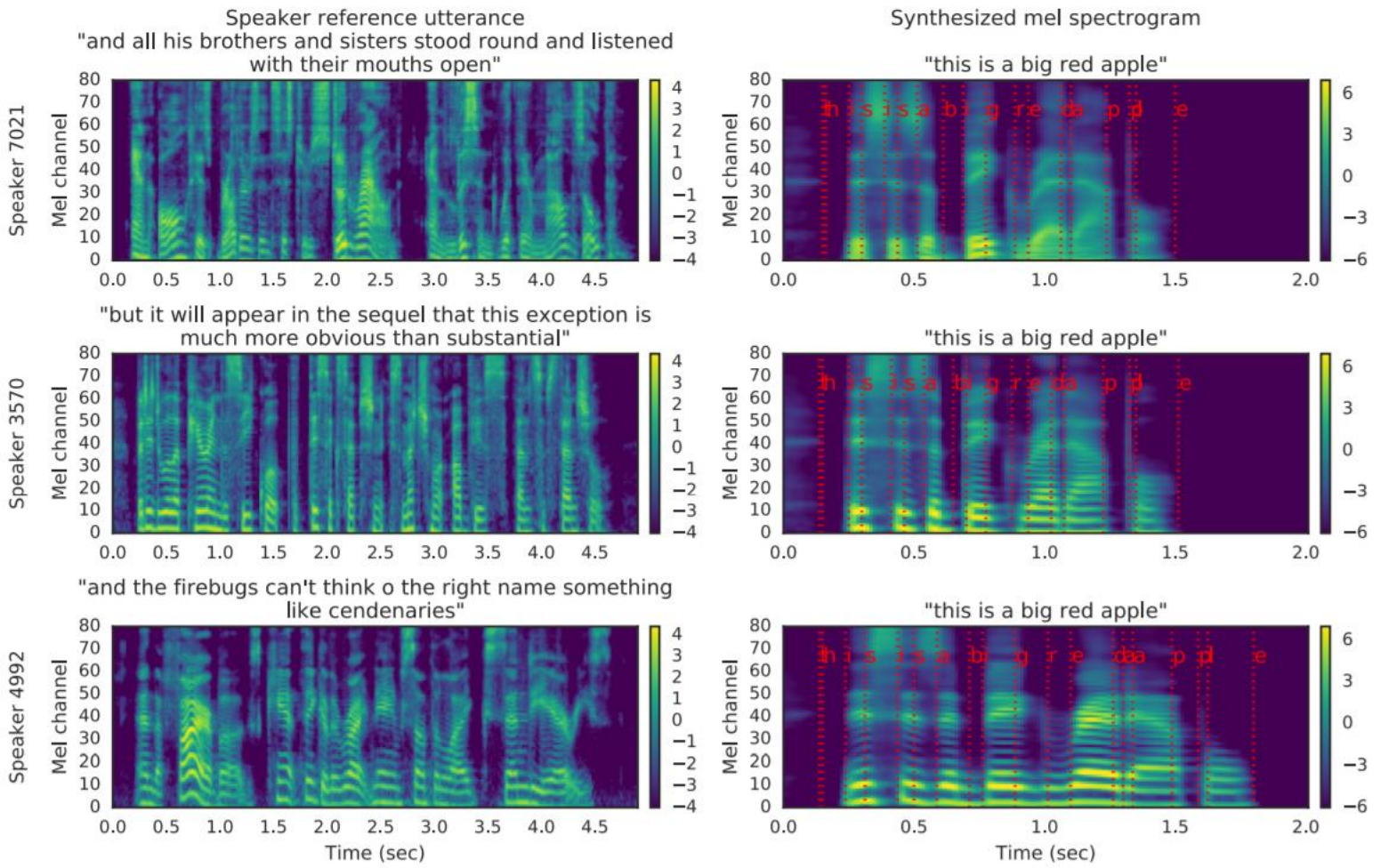


Figure 2: Example synthesis of a sentence in different voices using the proposed system. Mel spectrograms are visualized for reference utterances used to generate speaker embeddings (left), and the corresponding synthesizer outputs (right). The text-to-spectrogram alignment is shown in red. Three speakers held out of the train sets are used: one male (top) and two female (center and bottom).

Table 4: Speaker verification EERs of different synthesizers on unseen speakers.

Synthesizer Training Set	Training Speakers	SV-EER on VCTK	SV-EER on LibriSpeech
Ground truth	–	1.53%	0.93%
VCTK	98	10.46%	29.19%
LibriSpeech	1.2K	6.26%	5.08%

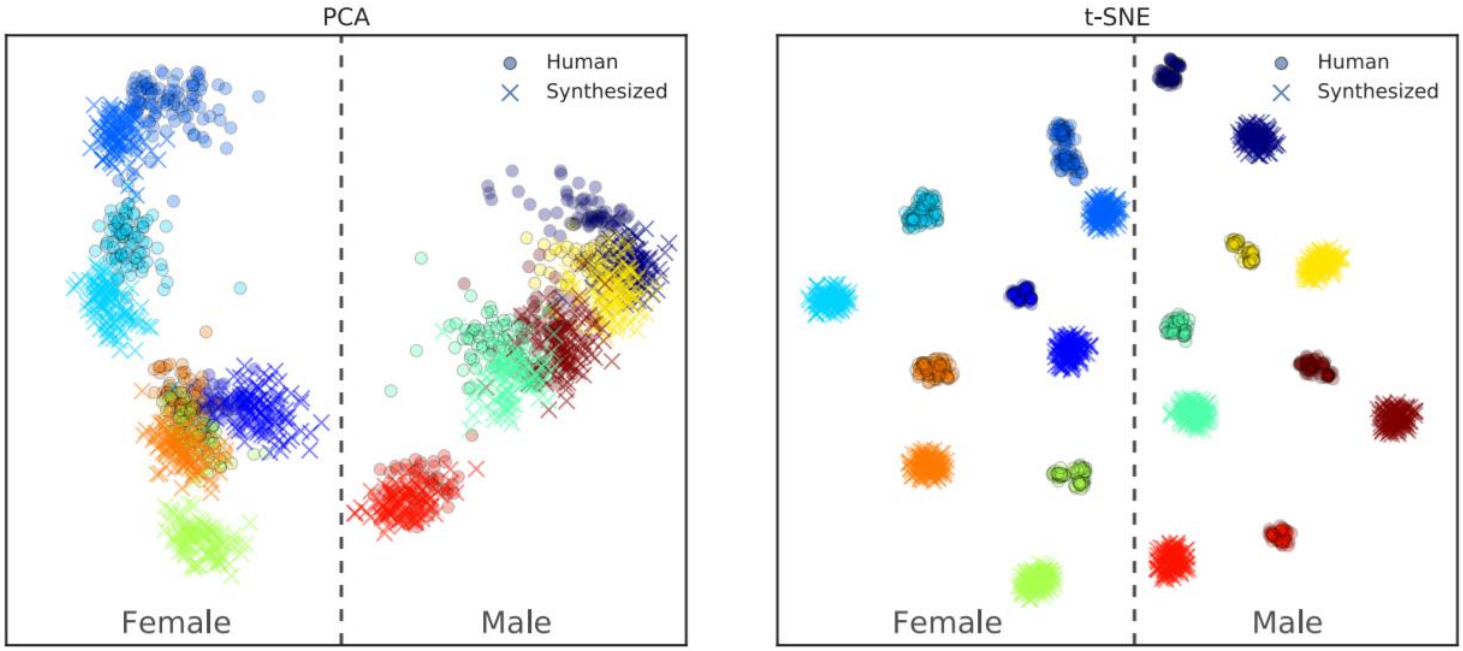


Figure 3: Visualization of speaker embeddings extracted from LibriSpeech utterances. Each color corresponds to a different speaker. Real and synthetic utterances appear nearby when they are from the same speaker, however real and synthetic utterances consistently form distinct clusters.

Table 6: Speech from fictitious speakers compared to their nearest neighbors in the train sets. Synthesizer was trained on LS Clean. Speaker Encoder was trained on LS-Other + VC + VC2.

Nearest neighbors in	Cosine similarity	SV-EER	Naturalness MOS
Synthesizer train set	0.222	56.77%	3.65 ± 0.06
Speaker Encoder train set	0.245	38.54%	

Analyzing Hidden Representations in End-to-End Automatic Speech Recognition Systems

Yonatan Belinkov and James Glass, **MIT**

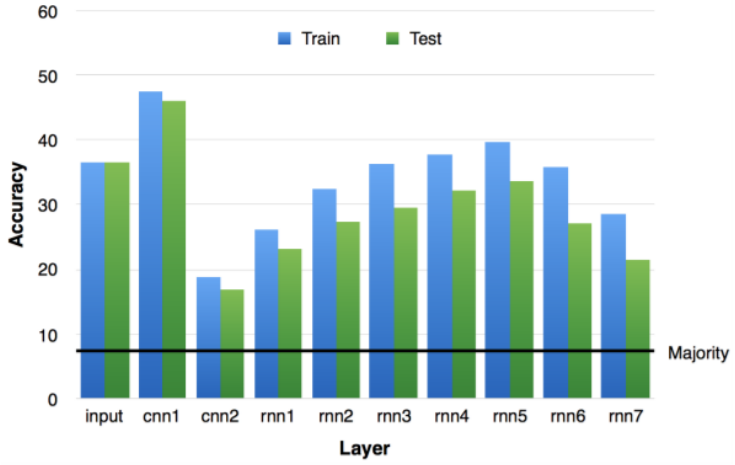
- CTC-based ASR system, Deep speech 2 architecture
- Trained on libriSpeech
- Train additional “prober” for each layer, for quick phone recognition on TIMIT.

(a) DeepSpeech2.

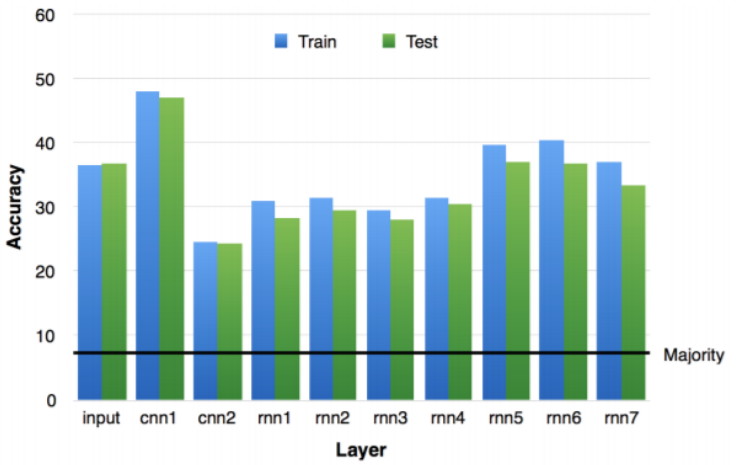
Layer	Type	Input Size	Output Size
1	cnn1	161	1952
2	cnn2	1952	1312
3	rnn1	1312	1760
4	rnn2	1760	1760
5	rnn3	1760	1760
6	rnn4	1760	1760
7	rnn5	1760	1760
8	rnn6	1760	1760
9	rnn7	1760	1760
10	fc	1760	29

(b) DeepSpeech2-light.

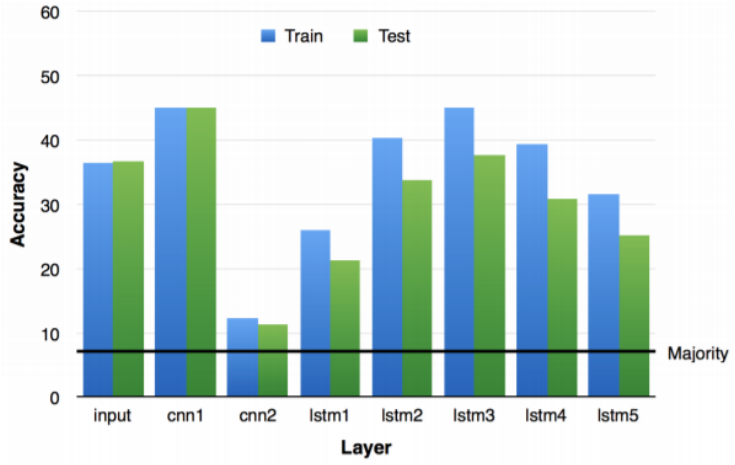
Layer	Type	Input Size	Output Size
1	cnn1	161	1952
2	cnn2	1952	1312
3	lstm1	1312	600
4	lstm2	600	600
5	lstm3	600	600
6	lstm4	600	600
7	lstm5	600	600
8	fc	600	29



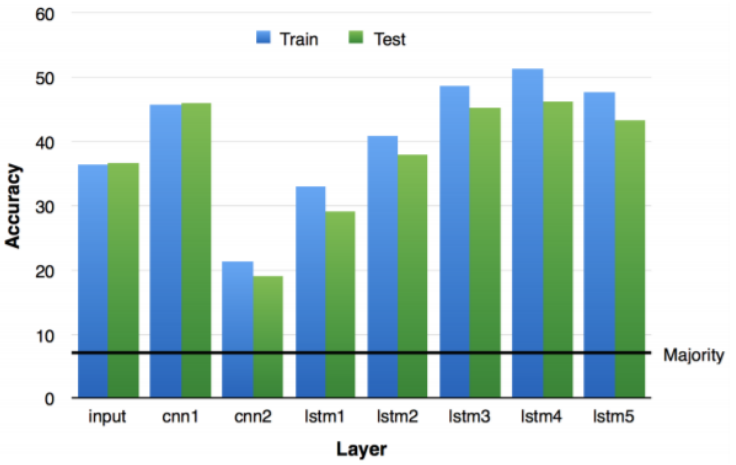
(a) DS2, w/ strides.



(b) DS2, w/o strides.



(c) DS2-light, w/ strides.



(d) DS2-light, w/o strides.

Figure 1: Frame classification accuracy using representations from different layers of DeepSpeech2 (DS2) and DeepSpeech2-light (DS2-light), with or without strides in the convolutional layers.

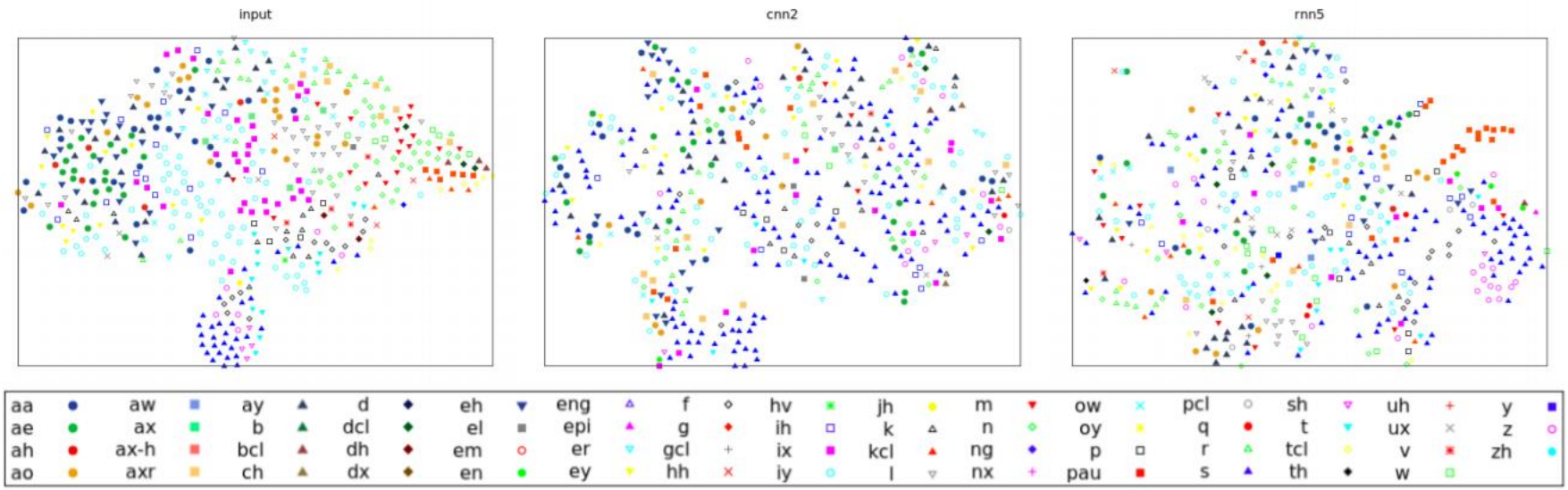


Figure 3: Centroids of frame representation clusters using features from different layers.

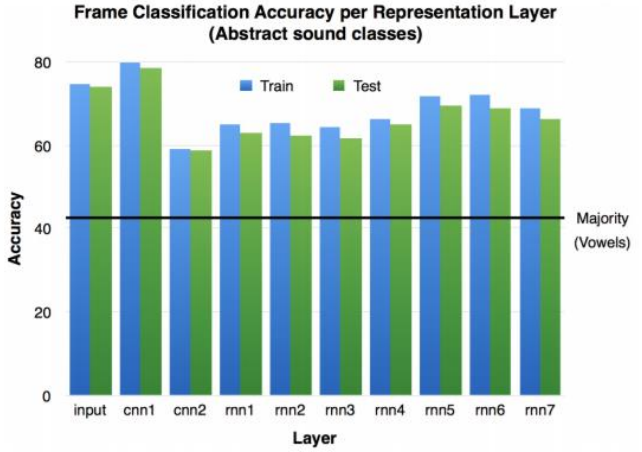


Figure 4: Accuracy of classification into sound classes using representations from different layers of DeepSpeech2.

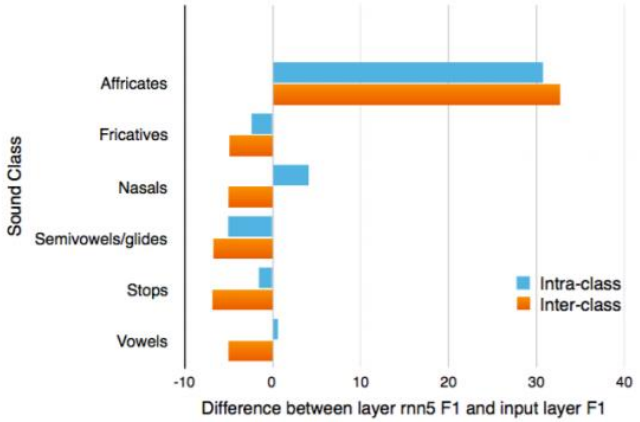
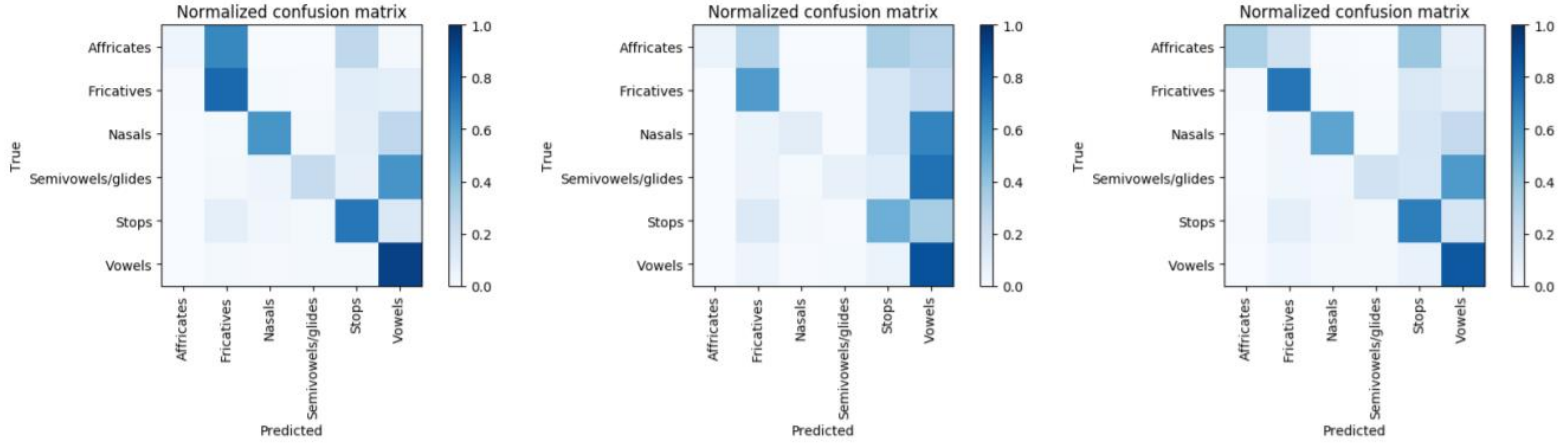


Figure 5: Difference in F1 score using representations from layer rnn5 compared to the input layer.



(a) input (b) cnn2 (c) rnn5

Figure 6: Confusion matrices of sound class classification using representations from different layers.

Houdini: Fooling Deep Structured Visual and Speech Recognition Models with Adversarial Examples
Moustapha Cisse et al. Facebook

Figure 1: We cause the network to generate a *minion* as segmentation for the adversarially perturbed version of the original image. Note that the original and the perturbed image are indistinguishable.

$$\tilde{x} = \underset{\tilde{x}:\|\tilde{x}-x\|_p \leq \epsilon}{\operatorname{argmax}} \ \ell\big(g_\theta(\tilde{x}), y\big)$$

- Traditionally an adverse sample is produced for individuals, by searching the most loss direction in the neighbor of a sample
- Design for impact performance of tasks

- Using a surrogate in the probability sense
- Attach deepspeech 2

$$\bar{\ell}_H(\theta, x, y) = \mathbb{P}_{\gamma \sim \mathcal{N}(0,1)} \Big[g_\theta(x, y) - g_\theta(x, \hat{y}) < \gamma \Big] \cdot \ell(\hat{y}, y)$$

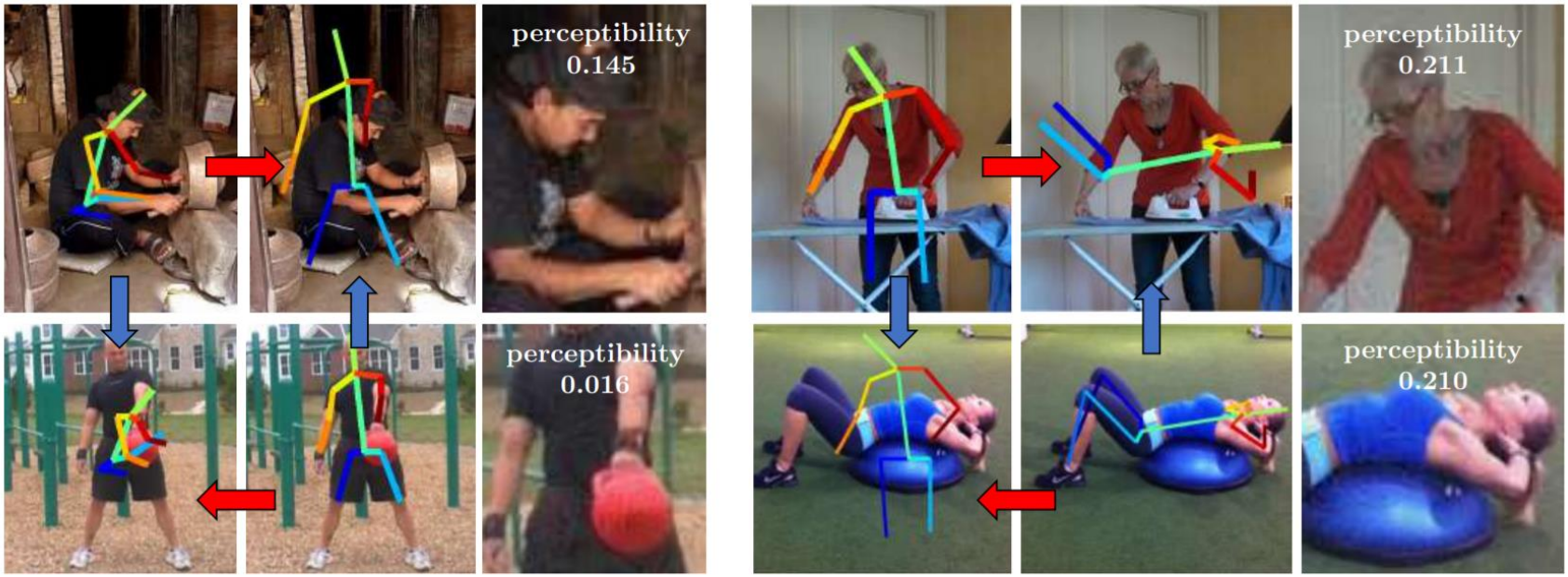


Figure 3: Examples of successful targetted attacks on a pose estimation system. Despite the important difference between the images selected, it is possible to make the network predict the wrong pose by adding an imperceptible perturbation. The images are part of the MPI dataset.

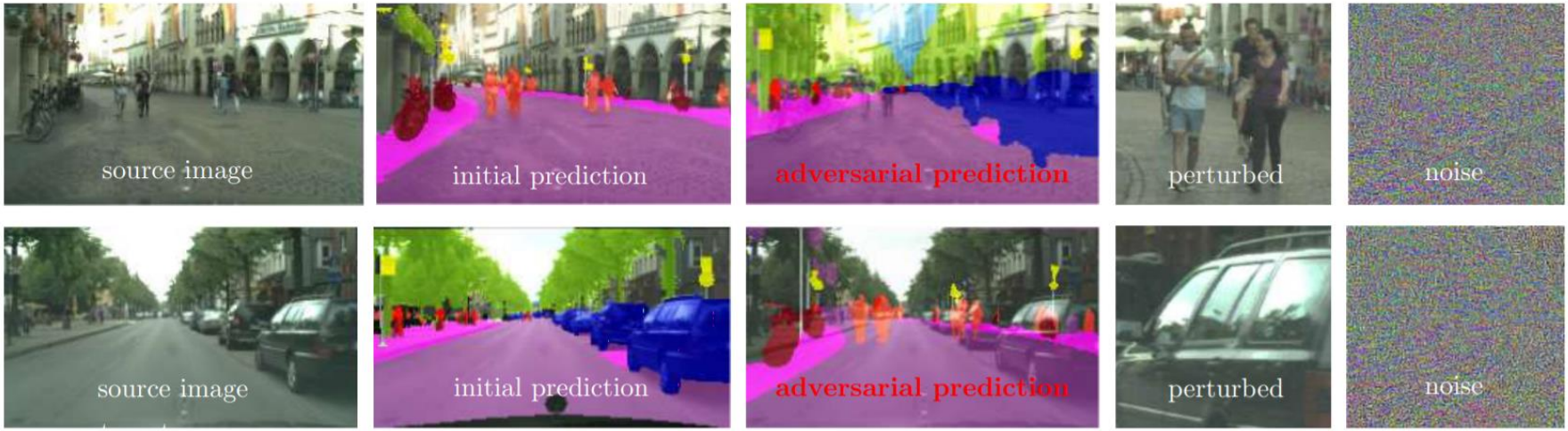
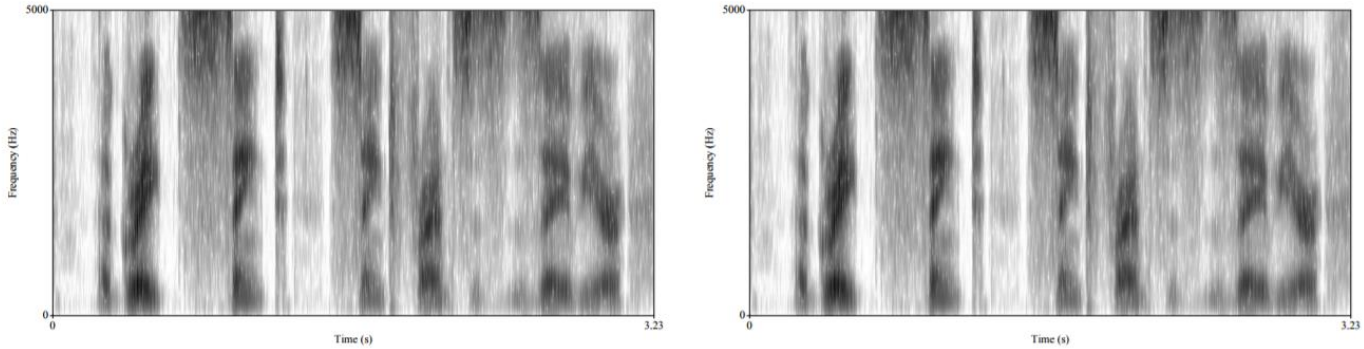


Figure 4: Targetted attack on a semantic segmentation system: switching target segmentation between two images from Cityscapes dataset [8]. The last two columns are respectively zoomed-in parts of the perturbed image and the adversarial perturbation added to the original one.

	$\epsilon = 0.3$		$\epsilon = 0.2$		$\epsilon = 0.1$		$\epsilon = 0.05$	
	WER	CER	WER	CER	WER	CER	WER	CER
CTC	68	9.3	51	6.9	29.8	4	20	2.5
Houdini	96.1	12	85.4	9.2	66.5	6.5	46.5	4.5

Figure 5: CER and WER in (%) for adversarial examples generated by both CTC and Houdini.



(a) a great saint saint francis zaviour (b) i great sinkt shink t frimsuss avir

Figure 6: The model's output for each of the spectrograms is located at the bottom of each spectrogram. The target transcription is: `A Great Saint Saint Francis Xavier.`

Groundtruth Transcription:
The fact that a man can recite a poem does not show he remembers any previous occasion on which he has recited it or read it.

G-Voice transcription of the original example:
The fact that a man can decide a poem does not show he remembers any previous occasion on which he has work cited or read it.

G-Voice transcription of the adversarial example:
The fact that I can rest I'm just not sure that you heard there is any previous occasion I am at he has your side it or read it.

Groundtruth Transcription:
Her bearing was graceful and animated she led her son by the hand and before her walked two maids with wax lights and silver candlesticks.

G-Voice transcription of the original example:
The bearing was graceful an animated she let her son by the hand and before he walks two maids with wax lights and silver candlesticks.

G-Voice transcription of the adversarial example:
Mary was grateful then admitted she let her son before the walks to Mays would like slice furnace filter count six.

Figure 8: Transcriptions from Google Voice application for original and adversarial speech segments.

What the eyes see and the ears hear, the mind believes. (Harry Houdini)

How about voice watermarking?

Fully Neural Network Based Speech Recognition on Mobile and Embedded Devices

Jinhwan Park et al. Seoul National Univeristy

- Full neural model: CTC linear RNN AM, character RNN LM, beam search

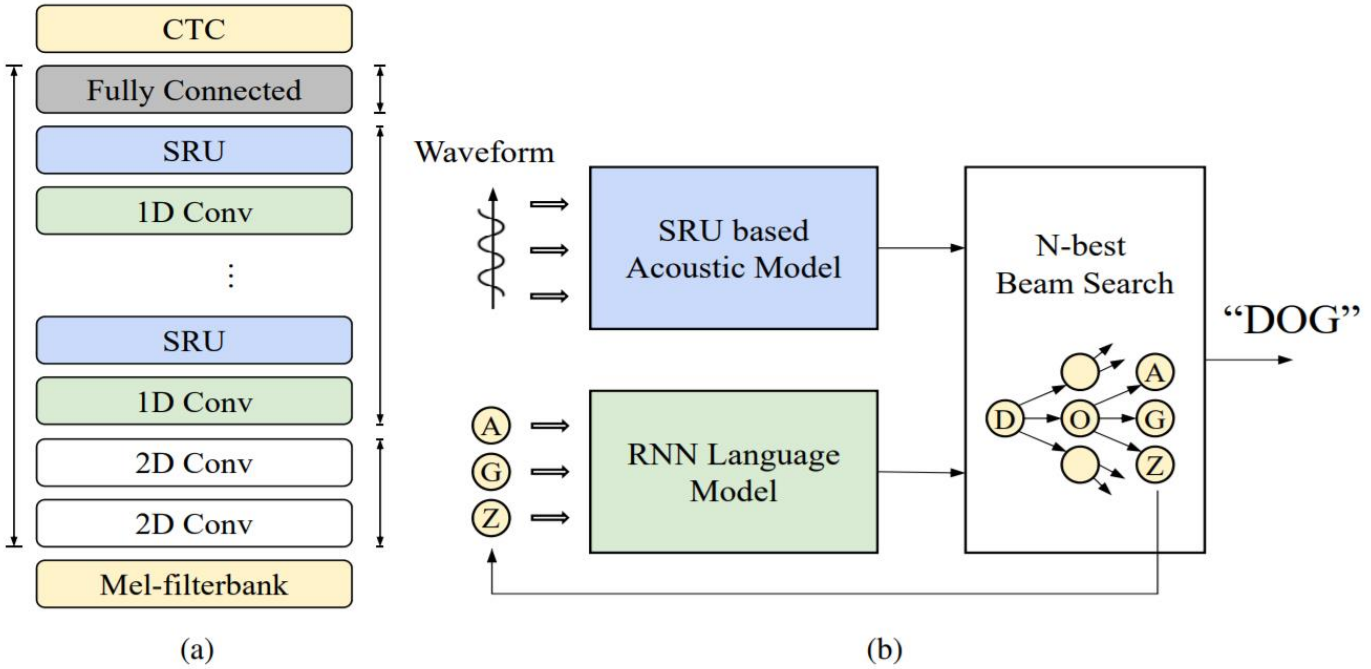


Figure 1: (a) The architecture of the neural network model used for acoustic modeling. (b) The system consists of RNN AM, RNN LM, and beam search decoding.

- Reduce DRAM access by parallel RNN

SRU:

$$
\begin{aligned}
\hat{\mathbf{x}}_t &= \mathbf{W}_z \mathbf{x}_t + \mathbf{b}_z, \\
\mathbf{f}_t &= \sigma(\mathbf{W}_f \mathbf{x}_t + \mathbf{b}_f), \\
\mathbf{o}_t &= \sigma(\mathbf{W}_o \mathbf{x}_t + \mathbf{b}_o), \\
\mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + (1 - \mathbf{f}_t) \odot \hat{\mathbf{x}}_t, \\
\mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t) + (1 - \mathbf{o}_t) \odot \mathbf{x}_t
\end{aligned} \tag{1}
$$

i-SRU:

$$
\begin{aligned}
\hat{\mathbf{x}}_t &= \tanh(\mathbf{W}_z \mathbf{x}_t + \mathbf{b}_z), \\
\mathbf{f}_t &= \sigma(\mathbf{W}_f \mathbf{x}_t + \mathbf{b}_f), \\
\mathbf{i}_t &= \sigma(\mathbf{W}_i \mathbf{x}_t + \mathbf{b}_i), \\
\mathbf{o}_t &= \sigma(\mathbf{W}_o \mathbf{x}_t + \mathbf{b}_o), \\
\mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{x}}_t \\
\mathbf{h}_t &= \mathbf{o}_t \odot \mathbf{c}_t + (1 - \mathbf{o}_t) \odot \mathbf{x}_t
\end{aligned} \tag{2}
$$

Table 1: WER and CER in percentage on WSJ eval92 test set. Decoding is conducted with RNN CLM and HCLM.

		Greedy		CLM		HCLM	
Model	**Params.**	**CER**	**WER**	**CER**	**WER**	**CER**	**WER**
6x800 SRU	10.62M	26.94	82.56	13.24	29.68	7.94	15.41
6x700 i-SRU	10.92M	12.70	45.22	7.04	18.90	4.90	12.27
6x800 SRU, 1-D conv	10.69M	6.06	22.16	3.48	9.53	**1.97**	**4.90**
6x700 i-SRU, 1-D conv	10.98M	**5.26**	**19.07**	**2.70**	**7.30**	2.01	**4.90**
6x1000 i-SRU, proj, 1-D conv	14.14M	5.85	21.60	3.00	7.80	2.27	5.17
4x600 LSTM	10.85M	7.29	24.88	5.35	14.27	3.70	8.75
4x600 LSTM, 1-D conv	10.88M	6.95	23.57	5.80	15.22	3.10	7.01
4x840 LSTM, proj, 1-D conv	12.01M	7.78	26.80	4.88	12.26	3.36	7.60
6x300 Gated ConvNet	16.38M	8.02	28.65	5.13	13.82	2.98	6.74
4x550 GILR-LSTM	11.34M	8.60	31.99	4.86	13.60	2.66	6.35
4x550 GILR-LSTM, 1-D conv	11.37M	7.15	26.06	4.44	11.92	2.38	5.45
bidirectional models							
6x400 i-SRU, 1-D conv	11.52M	**4.90**	**17.30**	**2.94**	**7.90**	**1.97**	**4.87**
4x350 LSTM	10.70M	5.88	20.17	3.46	9.41	2.57	5.89

Table 2: Comparision of the model with non-causal and causal 1-D convolutions. 1-D conv (-a, b) uses a past and b future time-steps to compute the output of the current time step.

	Greedy		CLM		HCLM	
Model	**CER**	**WER**	**CER**	**WER**	**CER**	**WER**
6x700 i-SRU, 1-D conv (-7, 7)	5.26	19.07	2.70	7.30	2.01	4.90
6x700 i-SRU, 1-D conv (-14, 0)	5.70	20.18	3.12	8.47	2.30	5.32
6x700 i-SRU, 1-D conv (-7, 0)	6.10	21.96	2.99	7.69	2.35	5.55
6x700 i-SRU, 1-D conv (-7, 1)	6.02	21.40	3.09	8.04	2.39	5.57
6x700 i-SRU, 1-D conv (-7, 2)	6.32	22.80	3.08	7.80	2.26	5.28

Table 3: WER and CER in percentage on WSJ eval92 test set when trained with additional data.

		Greedy		CLM		HCLM	
Model	**Params.**	**CER**	**WER**	**CER**	**WER**	**CER**	**WER**
6x700 i-SRU, 1-D conv	10.98M	4.13	18.02	2.54	**6.04**	1.51	3.73
6x1000 i-SRU, proj, 1-D conv	14.14M	**3.80**	14.70	**2.19**	6.20	**1.48**	**3.70**
4x600 LSTM, 1-D conv	10.88M	4.35	**13.90**	3.72	10.15	2.55	5.92
4x840 LSTM, proj, 1-D conv	12.01M	5.76	20.15	3.54	9.25	2.53	5.79
Deep Speech 2	100M	WER 3.60 with 5-gram LM					

Table 4: WER and CER on WSJ eval92 when word piece units are used.

	Greedy		WPLM	
Model	**CER**	**WER**	**CER**	**WER**
6x700 i-SRU, 1-D conv	7.37	17.95	6.73	10.50
4x600 LSTM, 1-D conv	9.34	22.56	8.47	15.64
6x700 i-SRU, 1-D conv, additional data	5.47	14.38	3.11	8.28
4x600 LSTM, 1-D conv, additional data	6.57	15.32	4.53	11.48

Table 5: Comparison of WER and CER on WSJ eval 92 according to downsampling in the word piece AMs.

	Greedy		WPLM	
Model	**CER**	**WER**	**CER**	**WER**
x2 in conv. layer	7.02	18.95	6.05	10.93
x4 in conv. layer	8.05	20.24	6.55	11.83
x2 in conv. layer, x2 in recurrent layer	7.37	17.95	6.00	10.50
x4 in conv. layer, x2 in recurrent layer	10.30	25.58	7.83	13.99

Table 6: WER and CER on Librispeech *test-clean* . The models are trained on LibriSpeech *train-clean-100* and *train-clean-360*.

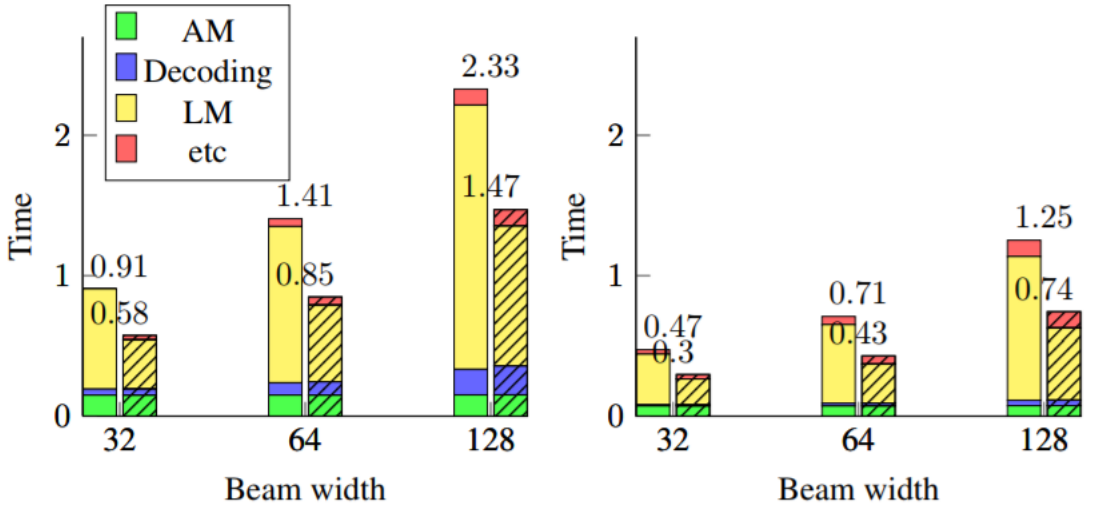
		Greedy		RNN LM	
Model	**Params.**	**CER**	**WER**	**CER**	**WER**
4x600 LSTM, character	10.85M	8.49	26.10	7.34	21.80
6x700 i-SRU, 1-D conv, character	10.98M	6.21	20.41	5.66	13.78
6x700 i-SRU, 1-D conv, word piece-500	11.30M	6.72	17.10	4.67	9.98
6x700 i-SRU, 1-D conv, word piece-1000	11.65M	6.62	16.16	4.42	9.61

Table 7: WER on Librispeech *test-clean* and *test-other*. The models are trained on all the LibriSpeech train set (960 hours).

Model	**Params.**	**test-clean**	**test-other**	**LM type**
6x700 i-SRU, 1-D conv	12M	9.02	23.60	RNN LM
12x1000 i-SRU, 1-D conv	36M	5.73	15.96	RNN LM
Gated ConvNet [21]	208M	4.8	14.5	4-gram LM
5-conv + 4x1024 bidirectional GRU [31]	75M	5.4	14.7	4-gram LM
Encoder-decoder [32]	150M	3.82	12.76	RNN LM

Table 8: Execution time of SRU-AM for 1 second of speech according to the number of parallelization steps.

Parallelization Step	1	2	4	8	16	32
Computation time	1.2129	0.6098	0.3065	0.2064	0.1524	0.1174



(a) Character-level model.

(b) Word piece-level model.

Beam	32	64	128
character	6.24	6.15	6.04
word piece	8.28	8.28	8.26
character (8-bit)	6.47	6.33	6.30
word piece (8-bit)	8.97	8.97	8.96

(c) WER with different beam width.

Figure 2: (a, b): Processing time of the speech recognition system for 1 second of speech on the single core ARM CPU. The time is evaluated on the WSJ eval92 dataset. The plot with dashed lines represents the computation time with 8-bit weights. (c): WERs when different beam width is used.

Unsupervised Cross-Modal Alignment of Speech and Text Embedding Spaces

Yu-An Chung et al. MIT

- Unpaired speech and text data, training something to make text words and spoken words aligned
- Skip-gram & Adversarial training
- Low resource ASR

- Speech2vec: segmentation, skip-gram, k-mean
  - This vector may contain both acoustic and semantic
- Align word vec and speech vec, however how to do that without supervision?

$$W^* = \underset{W \in \mathbb{R}^{d_2 \times d_1}}{\operatorname{argmin}} \|WX - Y\|^2$$

- Adversarial training

$$\mathcal{L}_D(\theta_D|W) = -\frac{1}{m}\sum_{i=1}^{m}\log P_{\theta_D}(\text{speech} = 1|Ws_i) - \frac{1}{n}\sum_{j=1}^{n}\log P_{\theta_D}(\text{speech} = 0|t_j),$$

$$\mathcal{L}_W(W|\theta_D) = -\frac{1}{m}\sum_{i=1}^{m}\log P_{\theta_D}(\text{speech} = 0|Ws_i) - \frac{1}{n}\sum_{j=1}^{n}\log P_{\theta_D}(\text{speech} = 1|t_j)$$

Table 2: Different configurations for training Speech2Vec to obtain the speech embeddings with decreasing level of supervision. The last column specifies whether the configuration is unsupervised.

Configuration	Speech2Vec training		Unsupervised
	How word segments were obtained	How embeddings were grouped together	
A & A^*	Forced alignment	Use word identity	✗
B	Forced alignment	k-means	✗
C	BES-GMM [35]	k-means	✓
D	ES-KMeans [36]	k-means	✓
E	SylSeg [37]	k-means	✓
F	Equally sized chunks	k-means	✓

Table 3: Accuracy on spoken word classification. $\text{EN}_{\text{ls}} - \text{en}_{\text{swc}}$ means that the speech and text embeddings were learned from the speech training data of English LibriSpeech and text training data of English SWC, respectively, and the testing audio segments came from English LibriSpeech. The same rule applies to Table 5 and Table 6. For the Word Classifier, $\text{EN}_{\text{ls}} - \text{en}_{\text{swc}}$ and $\text{EN}_{\text{swc}} - \text{en}_{\text{ls}}$ could not be obtained since it requires parallel audio-text data for training.

Corpora	$\text{EN}_{\text{ls}} - \text{en}_{\text{ls}}$	$\text{FR}_{\text{ls}} - \text{fr}_{\text{ls}}$	$\text{EN}_{\text{swc}} - \text{en}_{\text{swc}}$	$\text{DE}_{\text{swc}} - \text{de}_{\text{swc}}$	$\text{EN}_{\text{ls}} - \text{en}_{\text{swc}}$	$\text{EN}_{\text{swc}} - \text{en}_{\text{ls}}$
Nonalignment-based approach						
Word Classifier	89.3	83.6	86.9	80.4	–	–
Alignment-based approach with cross-modal supervision (parallel dictionary)						
A^*	25.4	27.1	29.1	26.9	21.8	23.9
Alignment-based approaches without cross-modal supervision (our approach)						
A	23.7	24.9	25.3	25.8	18.3	21.6
B	19.4	20.7	22.6	21.5	15.9	17.4
C	10.9	12.6	14.4	13.1	6.9	8.0
D	11.5	12.3	14.2	12.4	7.5	8.3
E	6.5	7.2	8.9	7.4	4.5	5.9
F	0.8	1.4	2.8	1.2	0.2	0.5

Table 4: Retrieved results of example audio segments that are considered incorrect in word classification. The match for each audio segment is marked in bold.

Rank	Input audio segments			
	beautiful	clever	destroy	suitcase
1	lovely	cunning	destroyed	bags
2	pretty	smart	**destroy**	suitcases
3	gorgeous	**clever**	annihilate	luggage
4	**beautiful**	crafty	destroying	briefcase
5	nice	wisely	destruct	**suitcase**

Can be applied in spoken term retrieval?

Table 6: Results on spoken word translation. We measure how many times one of the correct translations of the input audio segment is retrieved, and report precision@k for $k = 1, 5$.

Corpora	$\text{EN}_{\text{ls}} - \text{fr}_{\text{ls}}$		$\text{FR}_{\text{ls}} - \text{en}_{\text{ls}}$		$\text{EN}_{\text{swc}} - \text{de}_{\text{swc}}$		$\text{DE}_{\text{swc}} - \text{en}_{\text{swc}}$		$\text{EN}_{\text{ls}} - \text{de}_{\text{swc}}$		$\text{FR}_{\text{ls}} - \text{de}_{\text{swc}}$	
Average P@k	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5
Alignment-based approach with cross-modal supervision (parallel dictionary)												
A^*	47.9	56.4	49.1	60.1	40.2	51.9	43.3	55.8	34.9	46.3	33.8	44.9
Alignment-based approaches without cross-modal supervision (our approach)												
A	40.5	50.3	39.9	50.9	32.8	43.8	33.1	43.4	31.9	42.2	30.1	42.1
B	36.0	44.9	35.5	44.5	27.9	38.3	30.9	40.9	26.6	35.3	25.4	38.2
C	24.7	35.4	23.9	37.3	22.0	30.3	20.5	29.1	19.2	26.1	14.8	23.1
D	25.4	33.1	24.4	34.6	23.5	29.1	20.7	31.3	20.8	25.9	14.5	22.4
E	15.4	20.6	16.7	19.9	14.1	15.9	16.6	17.0	14.8	16.7	9.7	11.8
F	4.3	5.6	6.9	7.5	4.9	6.5	5.3	6.6	4.2	5.9	1.8	2.6
Majority Word Baseline												
Major-Word	1.1	1.5	1.6	2.2	1.2	1.5	2.0	2.7	1.1	1.5	1.6	2.2