Natural Language Processing with Deep Learning
CS224N/Ling284

Navdeep Jaitly
(njaitly@nvidia.com)

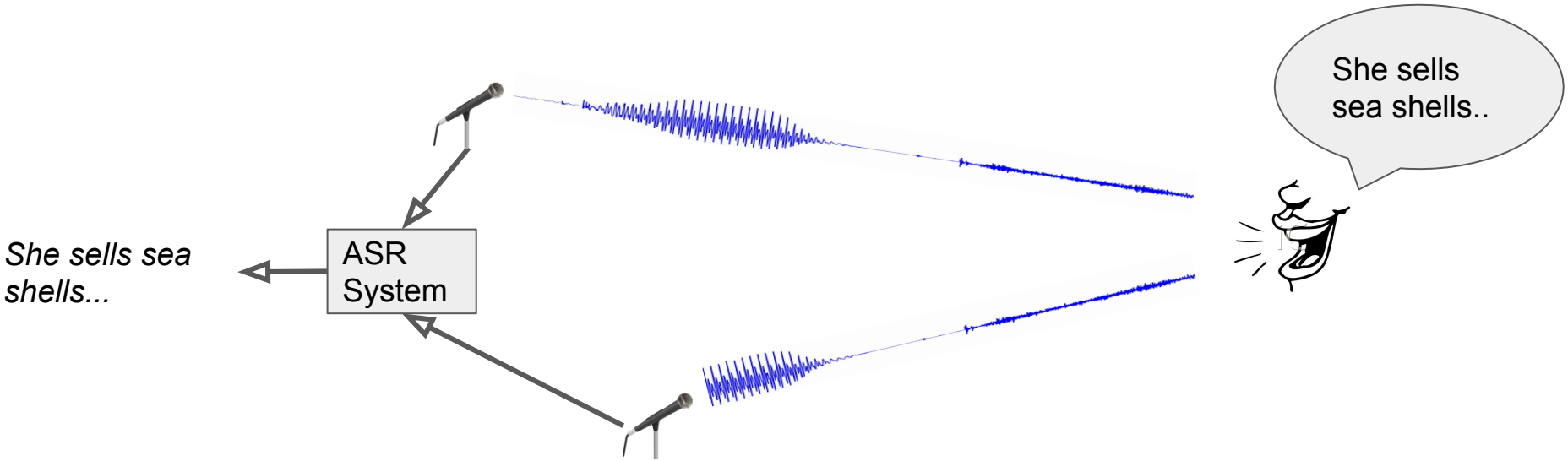
Lecture 12: End-to-end models for Speech Processing

Lecture Plan

1. Brief look at traditional speech recognition systems
2. Motivation for end-to-end models
3. Connectionist Temporal Classification (CTC)
4. Listen Attend and Spell (LAS) - a sequence-to-sequence based model for speech recognition
5. Improving LAS
    a. Convolutional models
    b. Addressing the target vocabulary problem
6. Online Sequence to Sequence models
7. Language Model Integration
8. Improving Decoding

Automatic Speech Recognition (ASR)

- Converting audio signal captured to the underlying textual representation



Why ASR ?

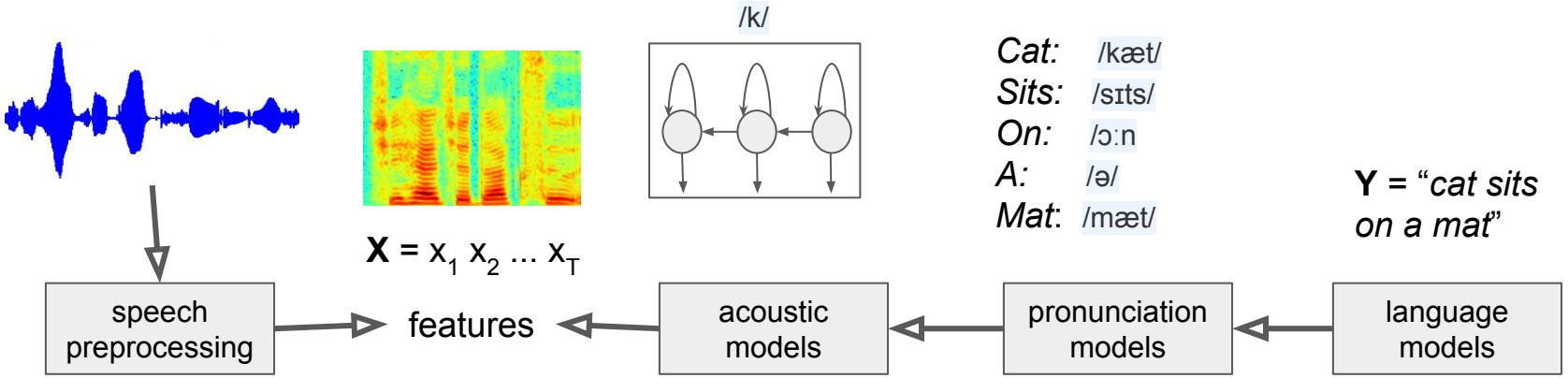
- Speech is a natural interface for human communication
  - Hands free communication.
  - No need to learn new techniques
- Applications are endless
  - Controlling simple devices -- cars, homes, handhelds (*OK! Google!*)
  - Interacting with intelligent devices -- chat bots, call center help desks, etc





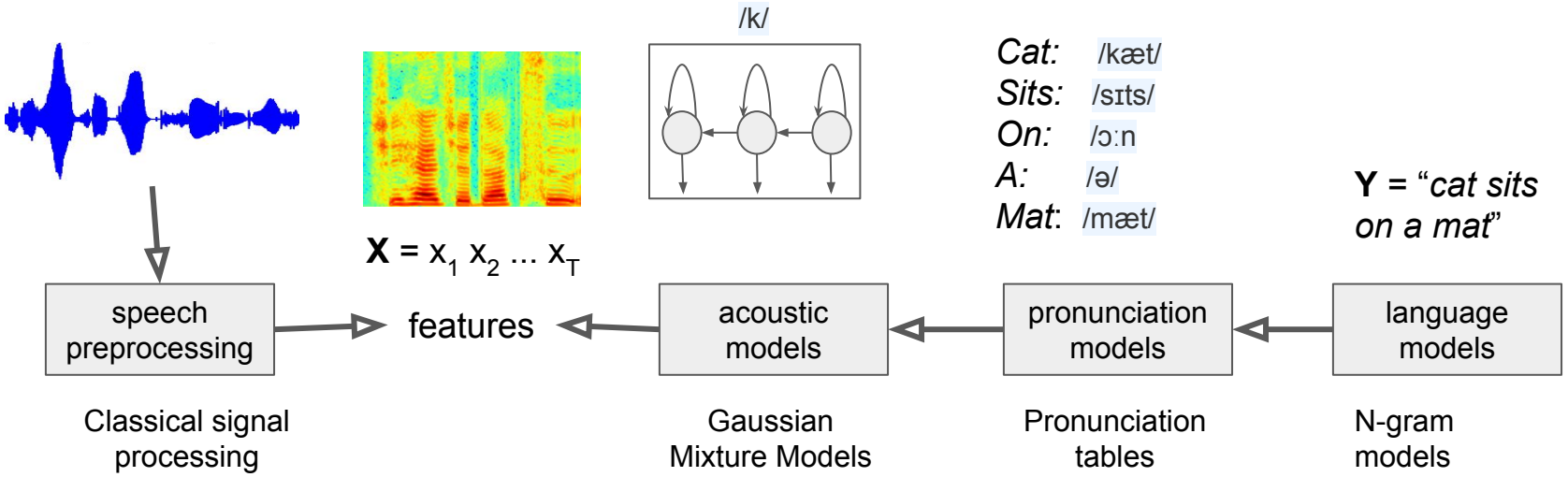
Speech Recognition -- the classical way

- Building a statistical model of speech starting from text sequences **Y**= $y_1 y_2 \ldots y_L$ to audio features **X** = $x_1 x_2 \ldots x_T$



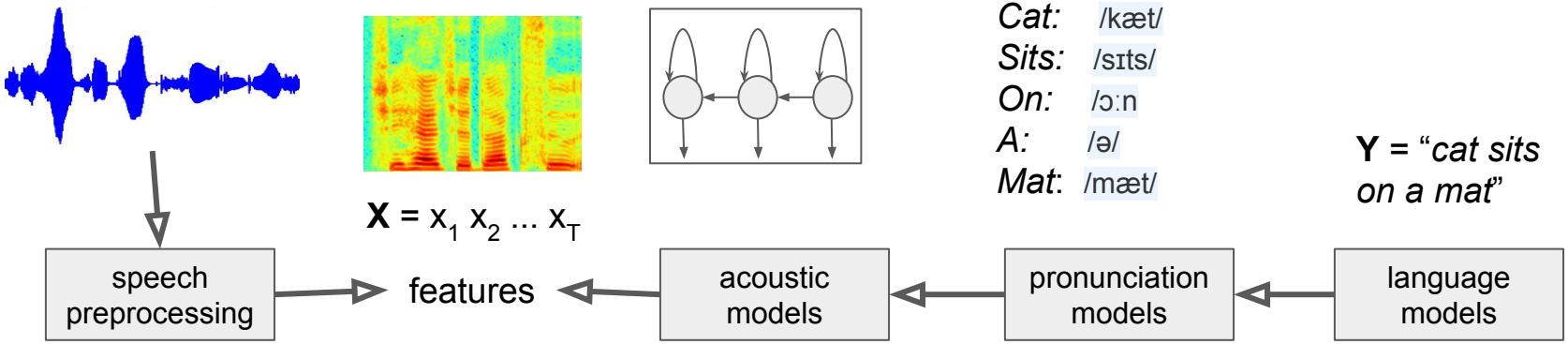
Speech Recognition -- the classical way

- Different statistical models used in different components



Speech Recognition -- the classical way

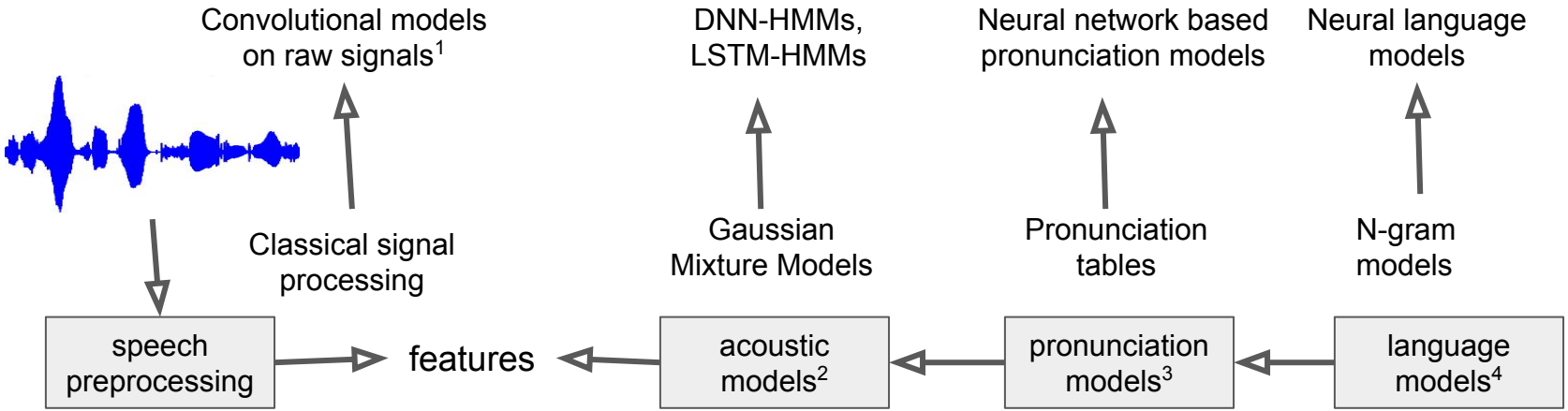
- Inference: Given audio features **X** = $x_1 x_2 \ldots x_T$ infer most likely text sequence **Y*** = $y_1 y_2 \ldots y_L$ that caused the audio features



$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} p(\mathbf{X}|\mathbf{Y})\, p(\mathbf{Y})$$

Speech Recognition -- the neural network invasion

- Each of the components seems to be better off with a neural network



1. Jaitly, Navdeep, and Geoffrey Hinton. "Learning a better representation of speech soundwaves using restricted boltzmann machines." *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011.
2. Hinton, Geoffrey, et al. "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups." *IEEE Signal Processing Magazine* 29.6 (2012): 82-97.
3. Rao, Kanishka, et al. "Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks." *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015.
4. Mikolov, Tomas, et al. "Recurrent neural network based language model." *Interspeech*. Vol. 2. 2010.

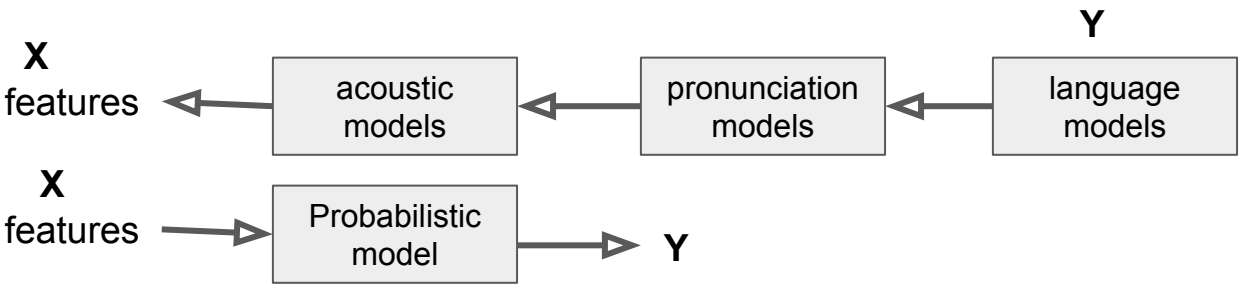
And yet ...

- Each component is trained independently, with a different objective!
- Errors in one component may not behave well with errors in another component
- Instead, lets train models that encompass all of these components together (end-to-end models)
  - Connectionist Temporal Classification (CTC)
  - Sequence to sequence (Listen Attend and Spell)

Treat end-to-end speech recognition as a modeling task

- Given audio **X** = $x_1 x_2 \ldots x_T$ and corresponding output text **Y**= $y_1 y_2 \ldots y_L$ where y $\in$ {a, b, c, d, … z, ?, !, …}
- **Y** is just a text sequence (transcript), **X** is the audio / processed spectrogram
- Perform speech recognition, by learning a probabilistic model p(**Y**|**X**)

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} p(\mathbf{Y}|\mathbf{X})$$

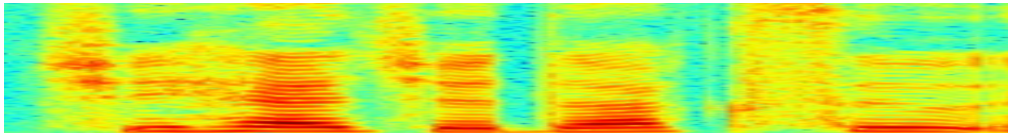


Connectionist Temporal Classification (CTC)

- CTC - a probabilitistic model p(**Y**|**X**), where
  - **X** = $x_1 x_2 \ldots x_T$,
  - **Y**= $y_1 y_2 \ldots y_L$
  - T>= L
- Has a specific structure that is suited for speech

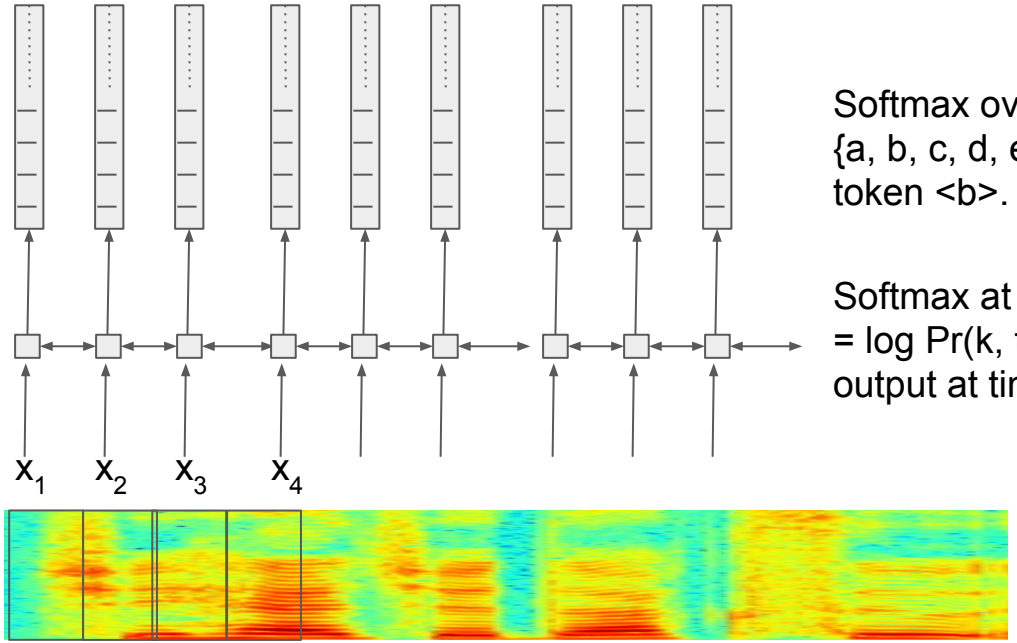
Y = This is a spectrogram

X =



- Graves, Alex, and Navdeep Jaitly. "Towards End-To-End Speech Recognition with Recurrent Neural Networks." *ICML*. Vol. 14. 2014.
- Amodei, Dario, et al. "Deep speech 2: End-to-end speech recognition in english and mandarin." *arXiv preprint arXiv:1512.02595* (2015).
- H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, “Learning acoustic frame labeling for speech recognition with recurrent neural networks,” in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2015.

Connectionist Temporal Classification



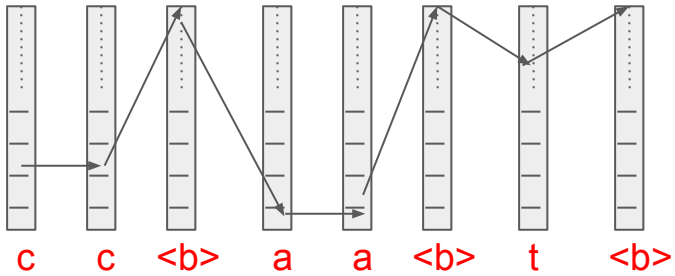
Softmax over vocabulary {a, b, c, d, e, f, .. z, ?, ., !,..} and extra token <b>.

Softmax at step, t, gives a score s(k,t) = log Pr(k, t|**X**) to category k in the output at time t.

- Graves, Alex, and Navdeep Jaitly. "Towards End-To-End Speech Recognition with Recurrent Neural Networks." *ICML*. Vol. 14. 2014.
- Amodei, Dario, et al. "Deep speech 2: End-to-end speech recognition in english and mandarin." *arXiv preprint arXiv:1512.02595* (2015).
- H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, “Learning acoustic frame labeling for speech recognition with recurrent neural networks,” in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2015.

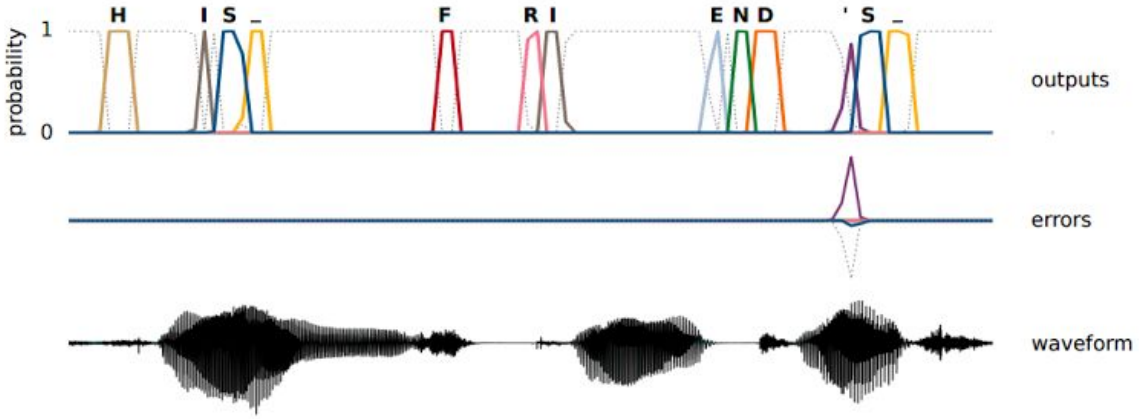
CTC - How frame predictions map to output sequences

- Repeated tokens are deduplicated
  - cc <b> aa <b> t <b>
- Any original transcript, maps to all possible paths in the duplicated space:
  - cc<b>aa<b>t<b> maps to cat
  - cc<b><b>a<b>t<b> maps to cat
  - ccccc<b>aaaaa<b>tttttt<b> maps to cat
  - ccccc<b>aaaaa<b>tttttt<b> maps to cat
- The score (log probability) of any path is the sum of the scores of individual categories at the different time steps
- The probability of any transcript is the sum of probabilities of all paths that correspond to that transcript



*Because of dynamic programming, it is possible to compute both the log probability p(**Y**|**X**) and its gradient exactly! This gradient can be propagated to neural network whose parameters can then be adjusted by your favorite optimizer!*

Connectionist Temporal Classification



Model learns to make peaky predictions!

CTC -- Some Examples

target: TO ILLUSTRATE THE POINT A PROMINENT MIDDLE EAST ANALYST IN WASHINGTON RECOUNTS A CALL FROM ONE CAMPAIGN

output: TWO ALSTRAIT THE POINT A PROMINENT MIDILLE EAST ANALYST IM WASHINGTON RECOUNCACALL FROM ONE CAMPAIGN

CTC - Somple Examples

target: T. W. A. ALSO PLANS TO HANG ITS BOUTIQUE SHINGLE IN AIRPORTS AT LAMBERT SAINT

output: T. W. A. ALSO PLANS TOHING ITS BOOTIK SINGLE IN AIRPORTS AT LAMBERT SAINT

CTC - Sompe Examples

target: THERE’S UNREST BUT WE’RE NOT GOING TO LOSE THEM TO DUKAKIS

output: THERE’S UNREST BUT WERE NOT GOING TO LOSE THEM TO DEKAKIS

CTC - Language Models

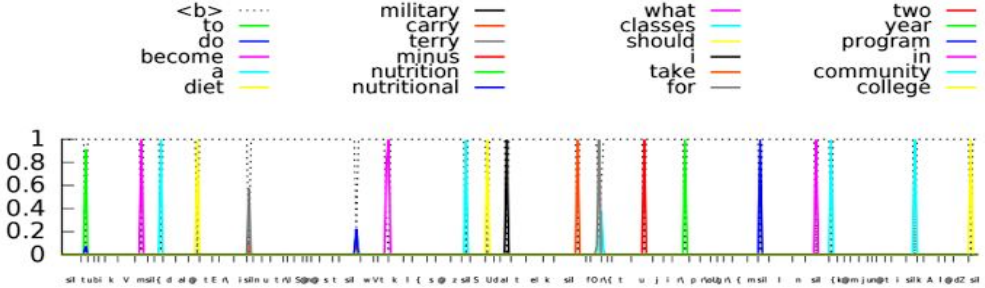
- Previous transcripts sounded correct, but clearly lacked the correct spelling and grammar
  - More training data can help, but eventually, a language model is required to fix these problems
  - With a simple language model rescoring, word error rate (WER) goes from 30.1% to 8.7%

- Google’s CTC implementation fixes these problems by integrating a language model into CTC during training

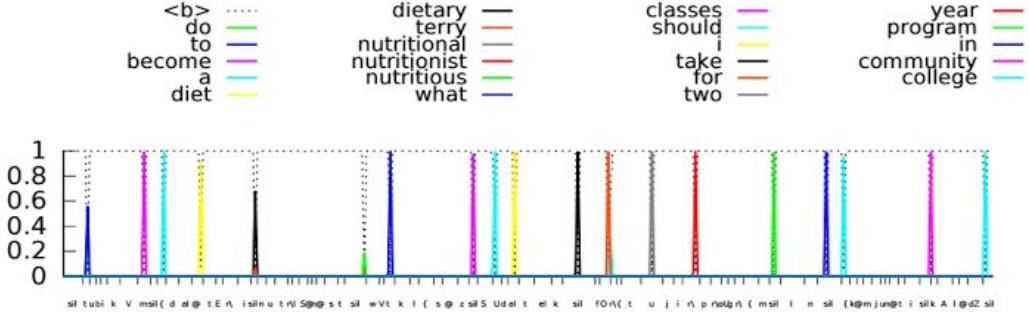
H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, “Learning acoustic frame labeling for speech recognition with recurrent neural networks,” in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2015.

CTC -- with word targets

7K Vocab



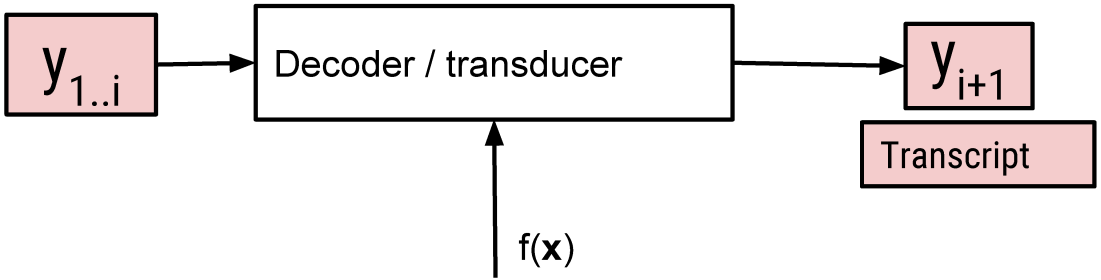
90K Vocab



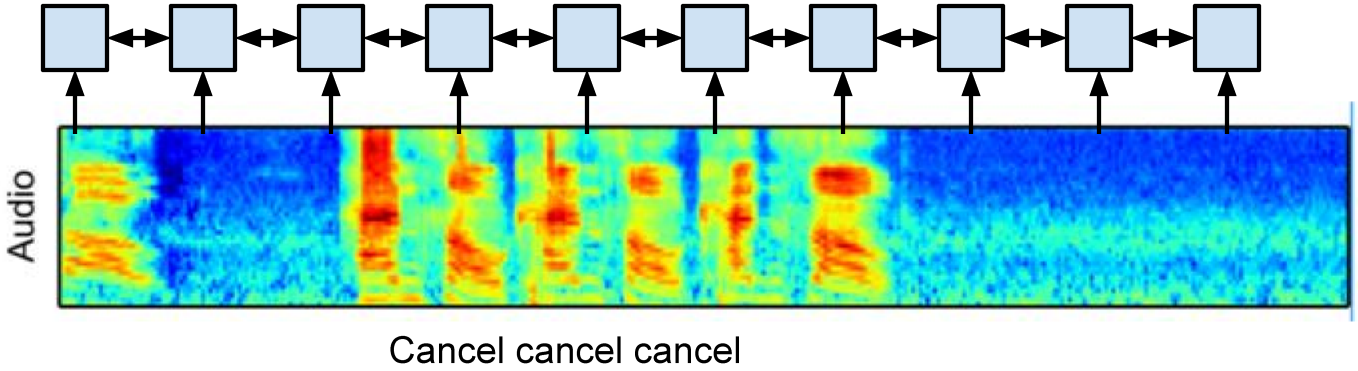
‘*To become a dietary nutritionist what classes should I take for a two year program in a community college*’

H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, “Learning acoustic frame labeling for speech recognition with recurrent neural networks,” in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2015.

Sequence-to-Sequence with attention for speech

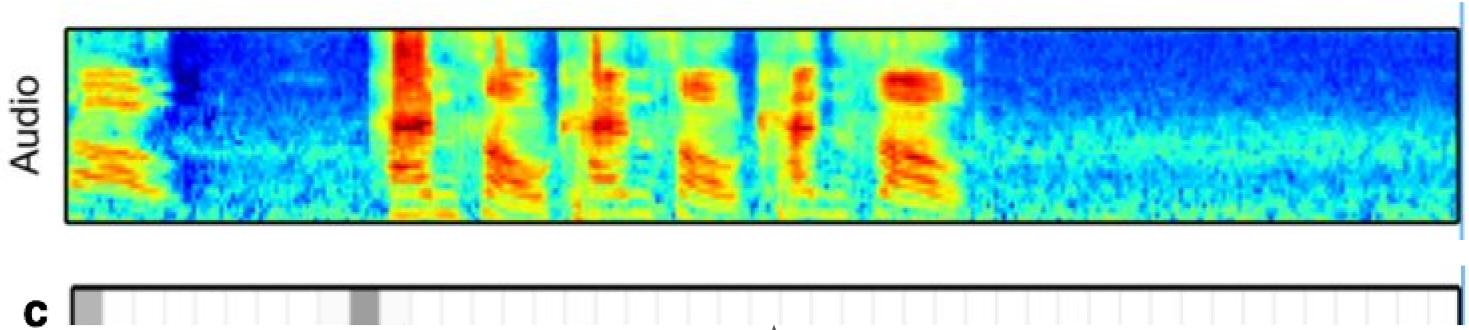


$p(y_{i+1}|y_{1..i}, x)$



William Chan, Navdeep Jaitly, Quoc Le, Oriol Vinyals. *Listen Attend and Spell*. ICASSP 2015.

Attention Example

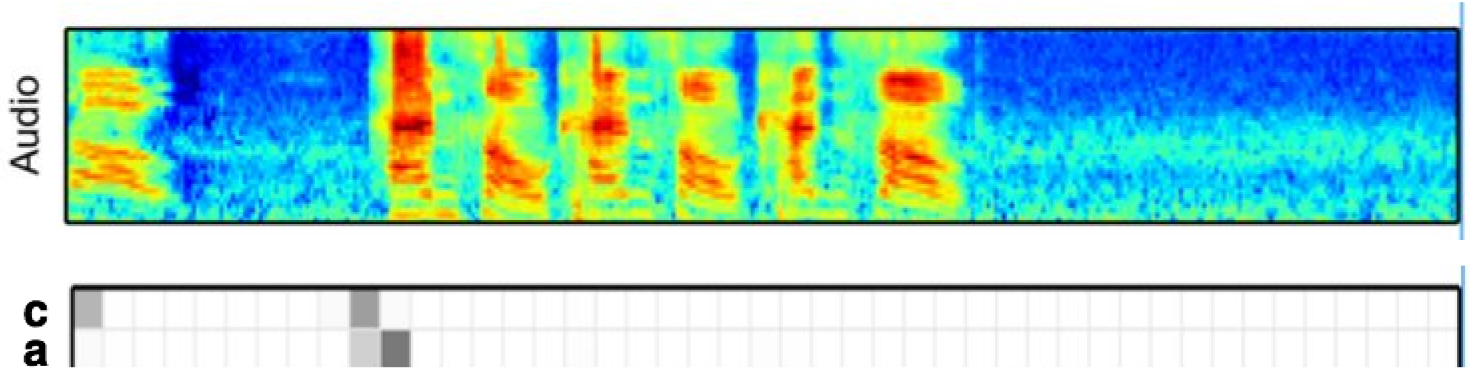


prediction derived from “attending” to segment of input

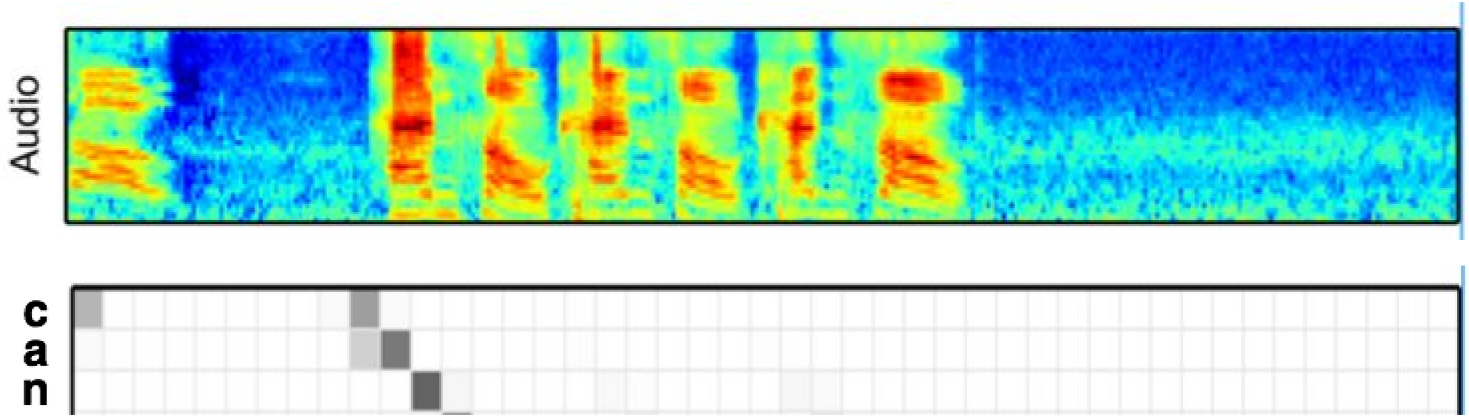
Attention vector - where the model thinks the relevant information is to be found

time

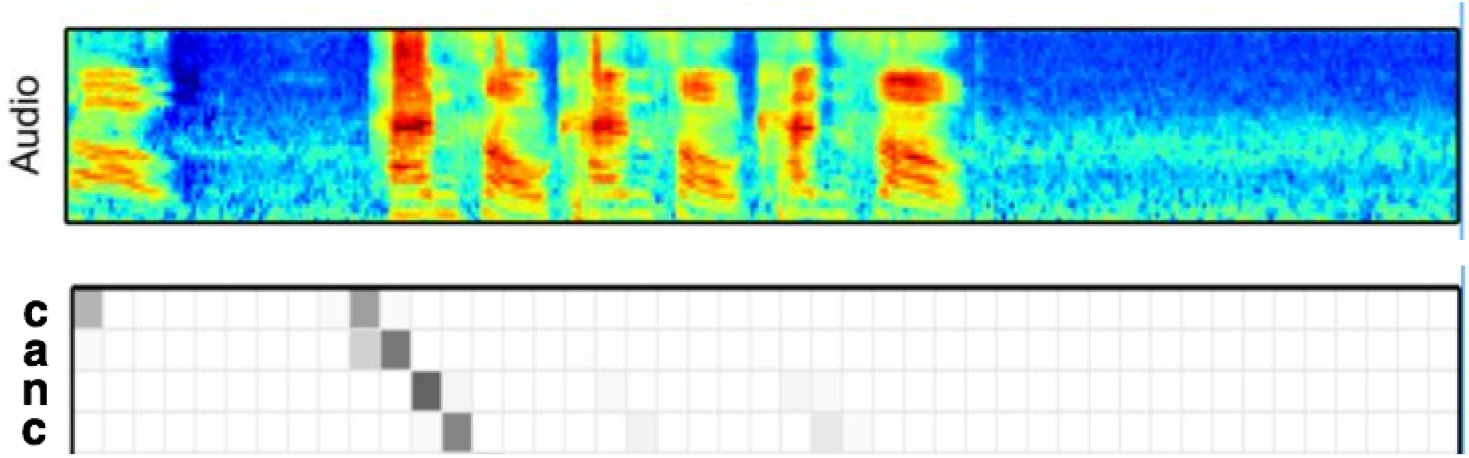
Attention Example



Attention Example

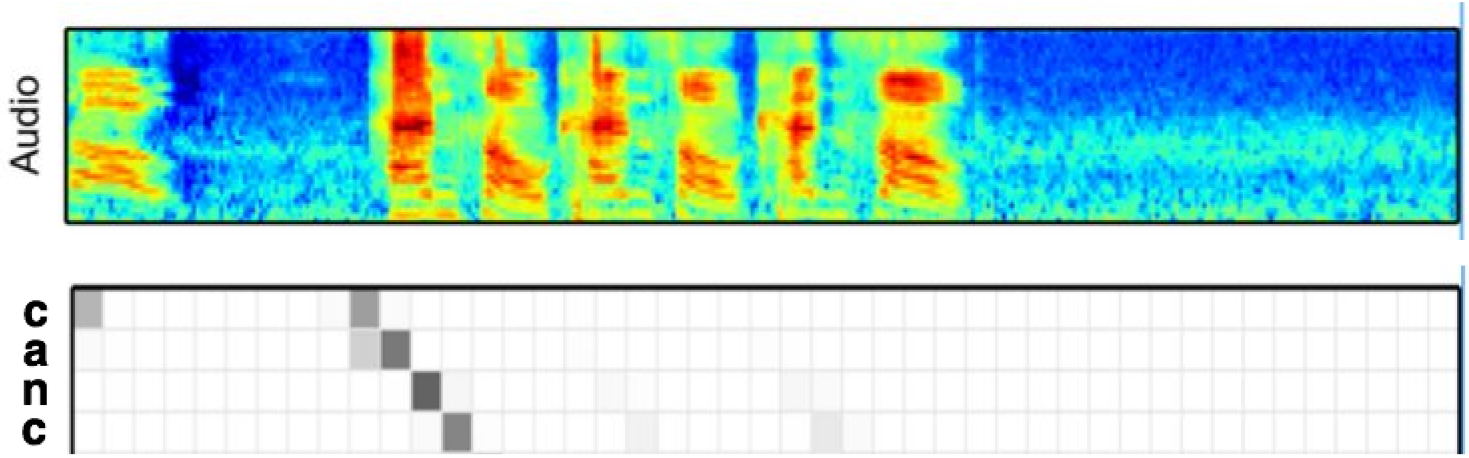


Attention Example



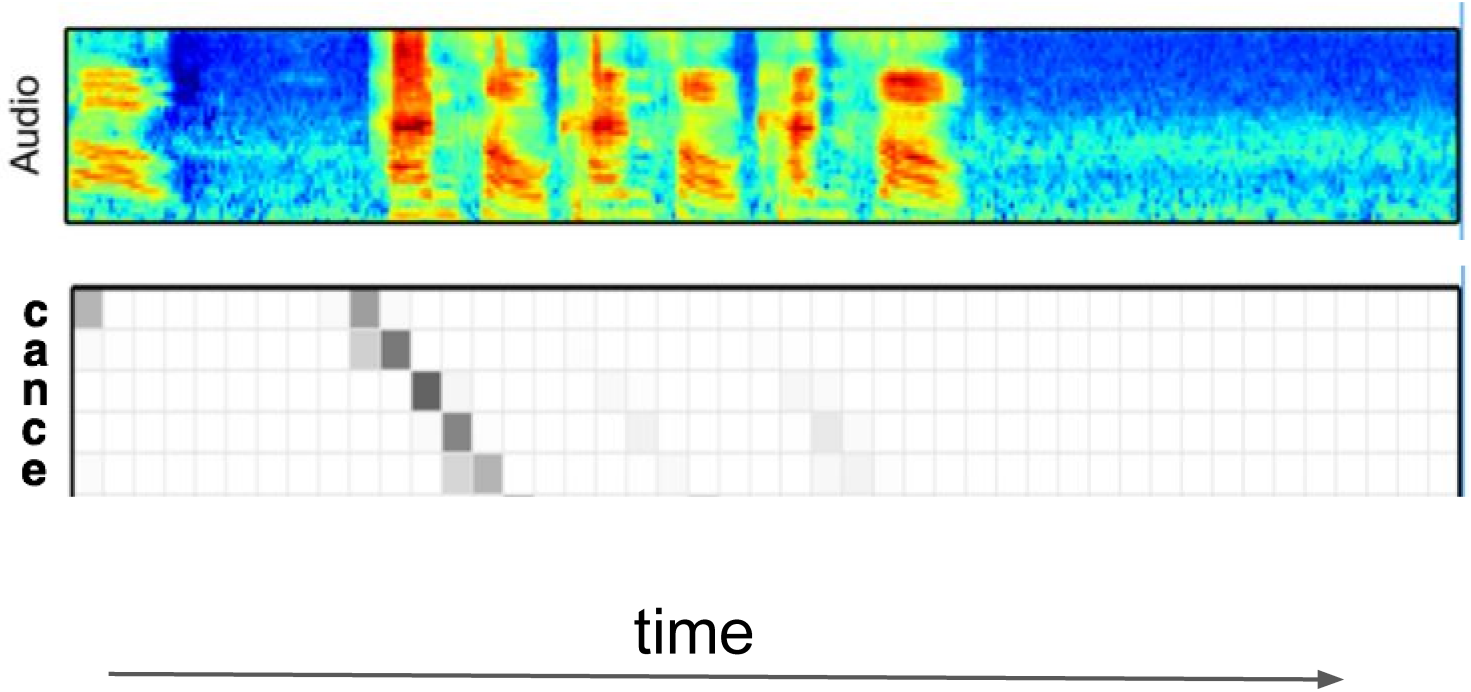
time

Attention Example

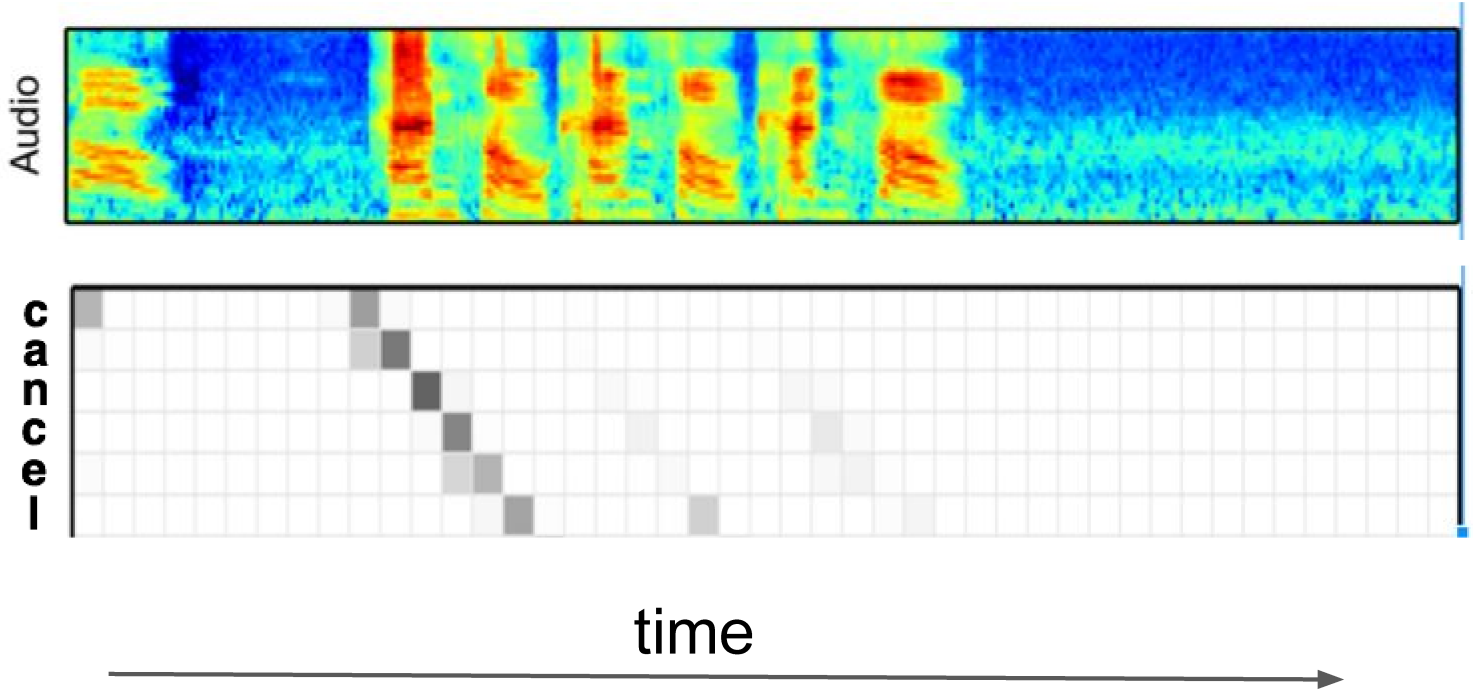


time

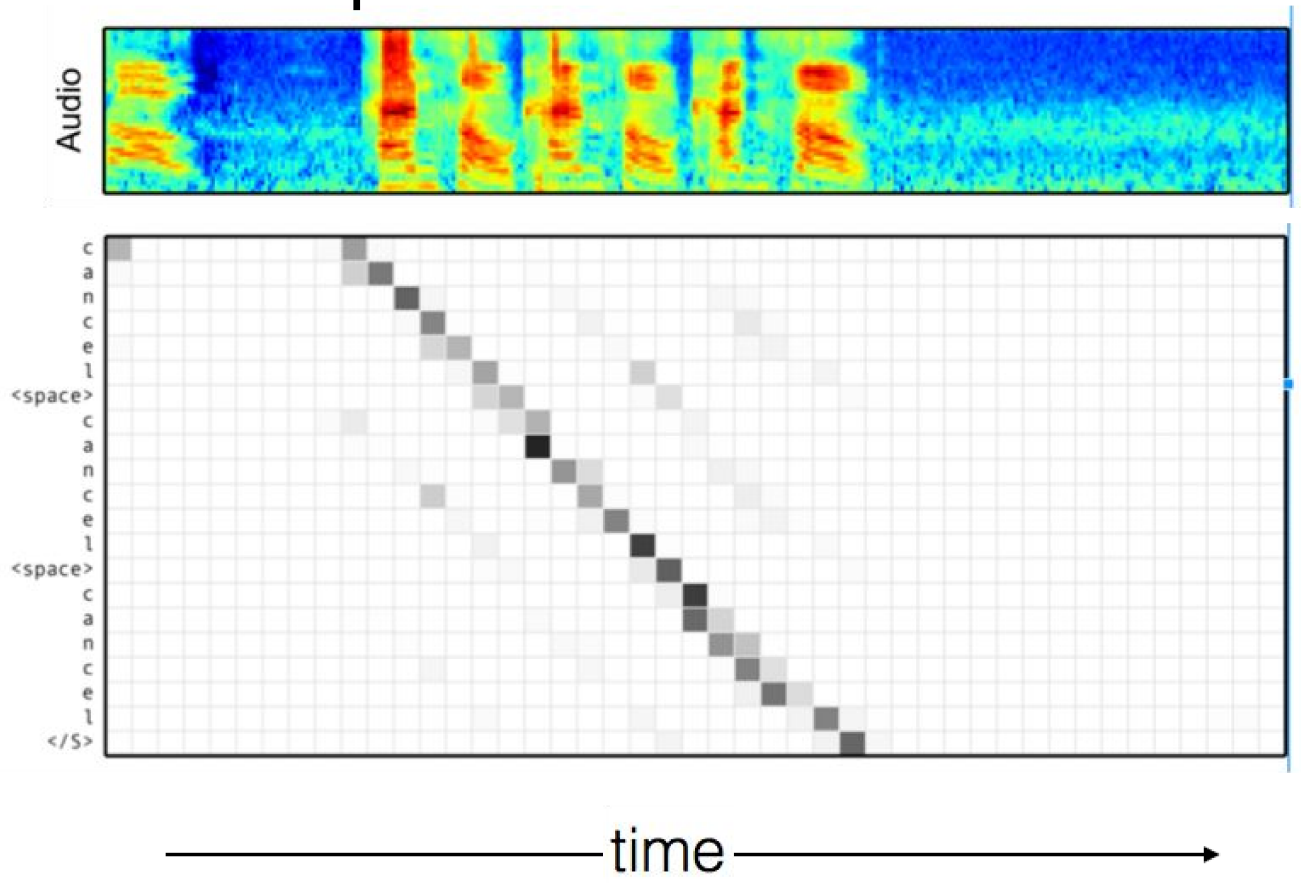
Attention Example



Attention Example

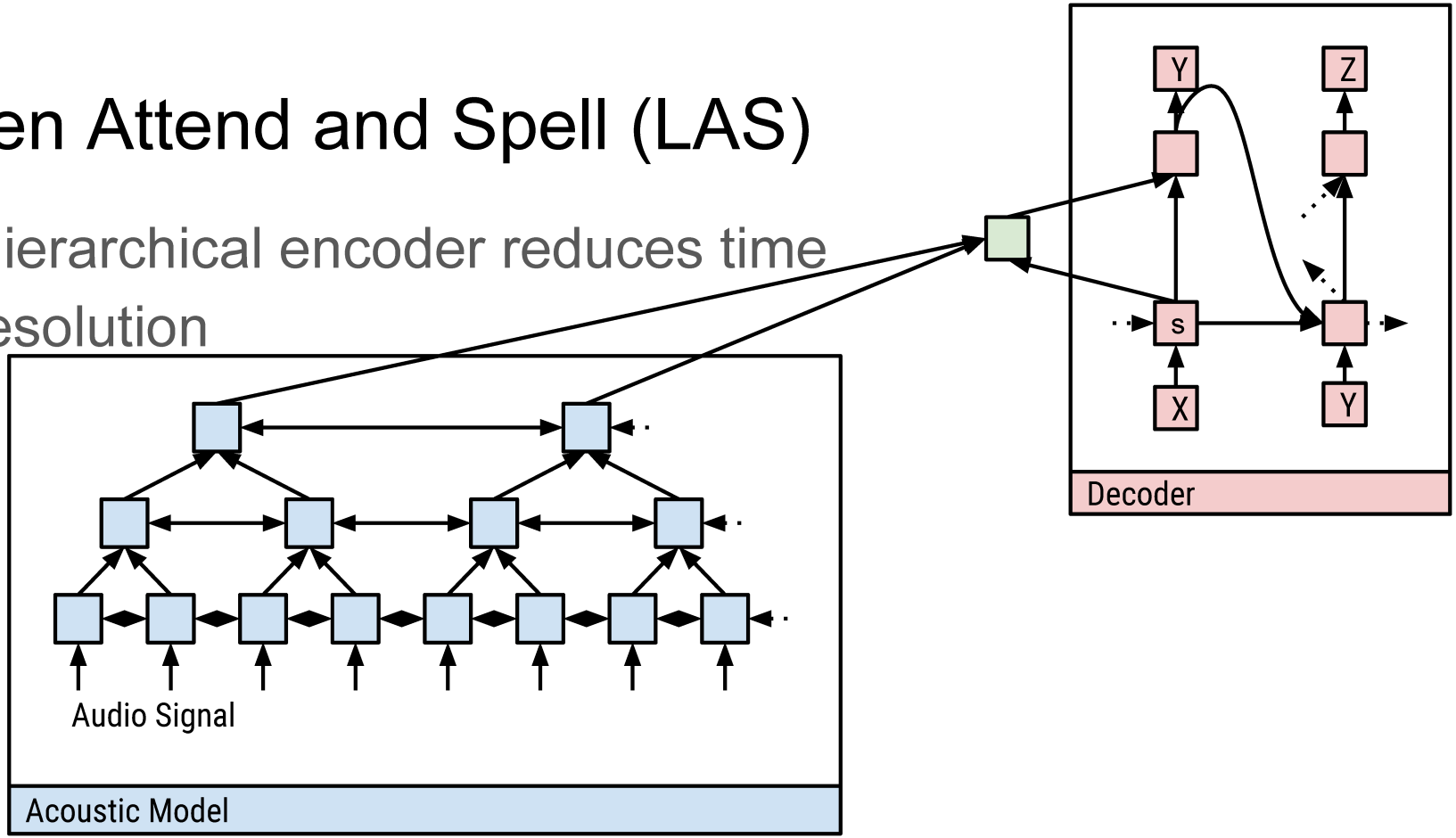


Attention Example



Listen Attend and Spell (LAS)

- Hierarchical encoder reduces time resolution



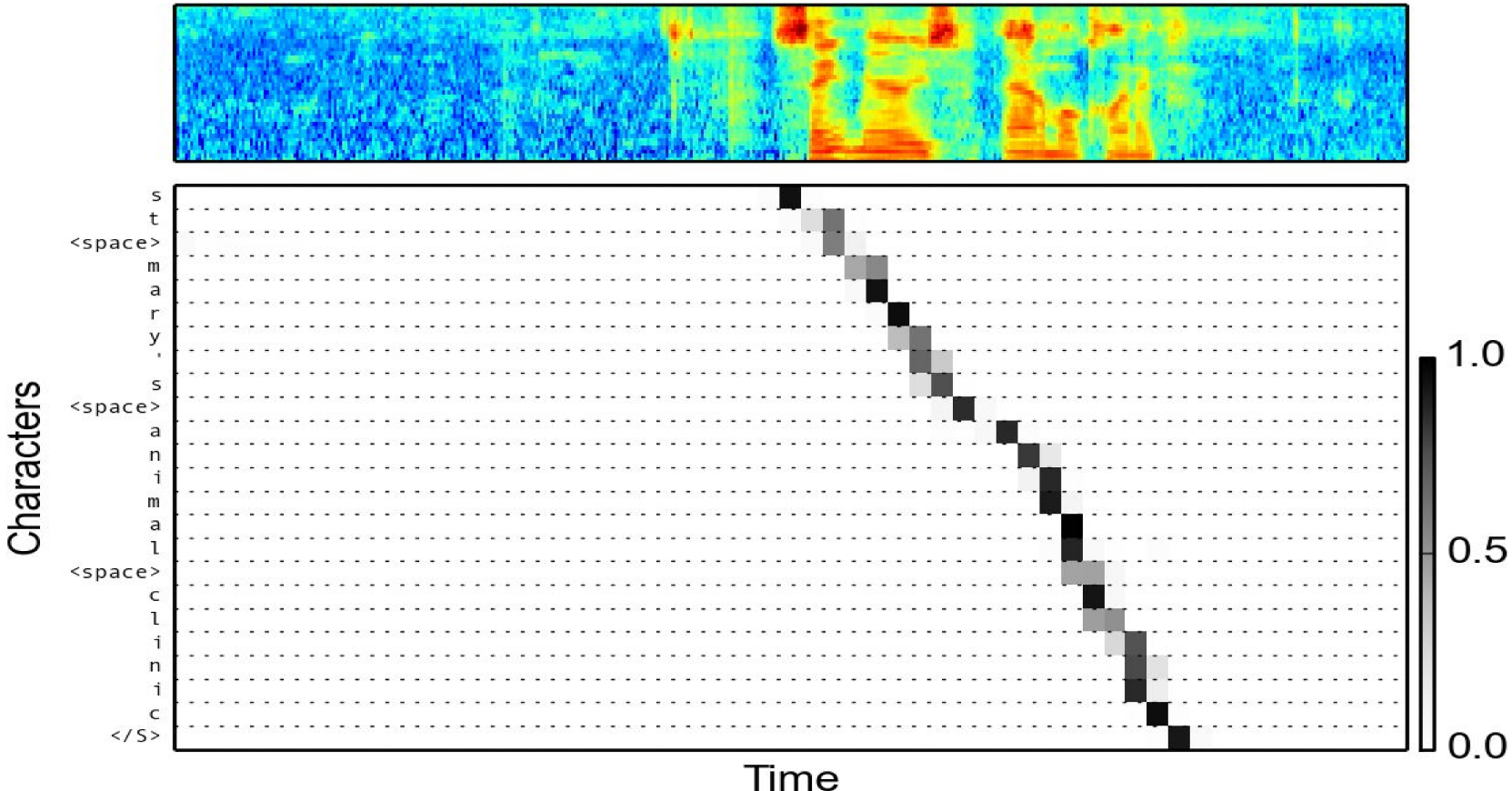
William Chan, Navdeep Jaitly, Quoc Le, Oriol Vinyals. *Listen Attend and Spell*. ICASSP 2015.

LAS highlights - Multimodal outputs

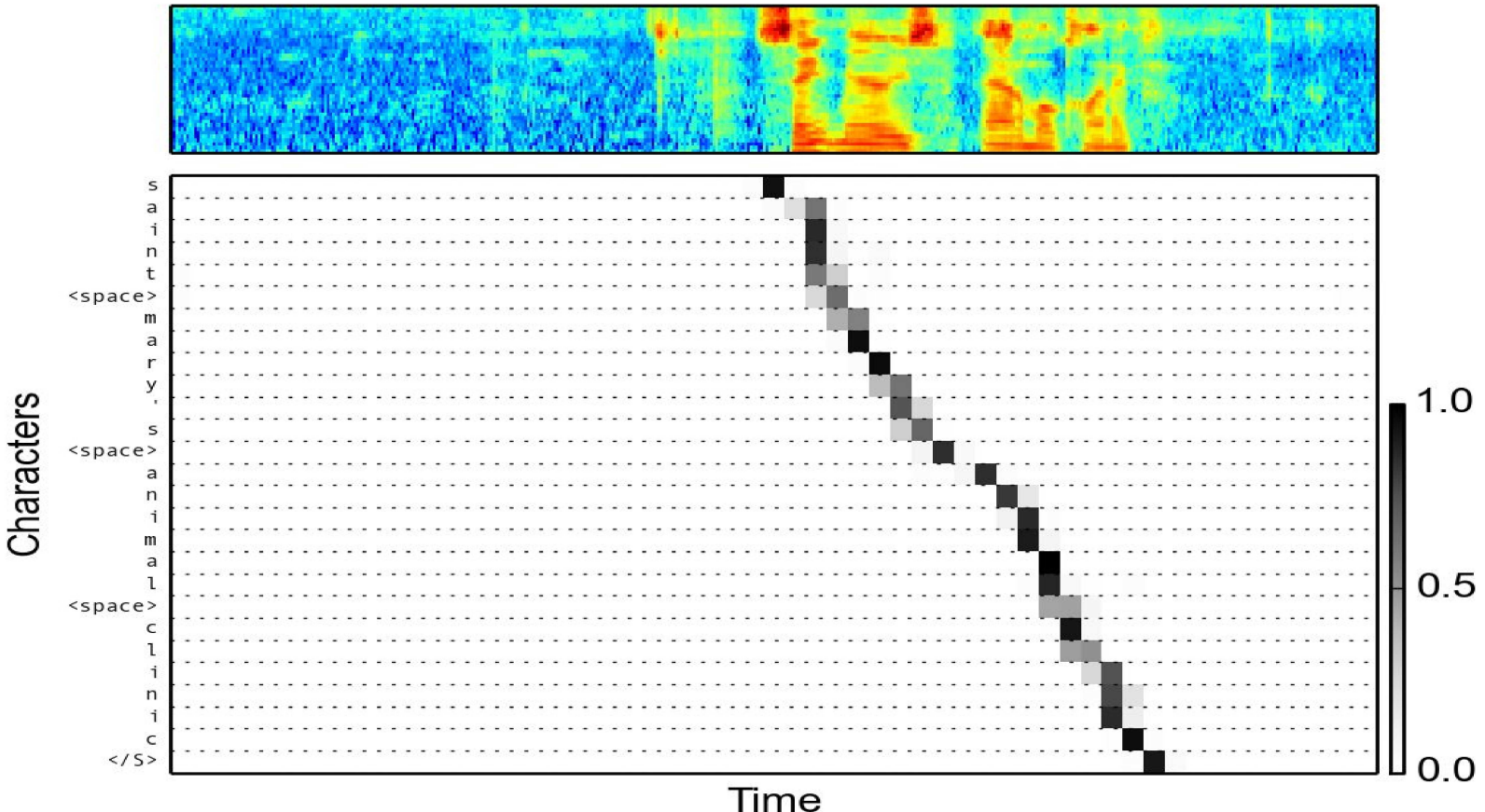
Beam	Text	LogProb	WER
Truth	call aaa roadside assistance	-	-
1	call aaa roadside assistance	-0.5740	0.00
2	call triple a roadside assistance	-1.5399	50.0
3	call trip way roadside assistance	-3.5012	50.0
4	call xxx roadside assistance	-4.4375	25.0

Very different outputs for same input!

LAS Highlights - Causality



LAS Highlights - Causality



LAS - Results

Model	Clean WER	Noisy WER
CLDNN-HMM (baseline)	8	8.9
Listen Attend and Spell (LAS)	14	16.5
LAS + external language model	10.3	12.0

Comparable to some of our best models without extensive engineering !

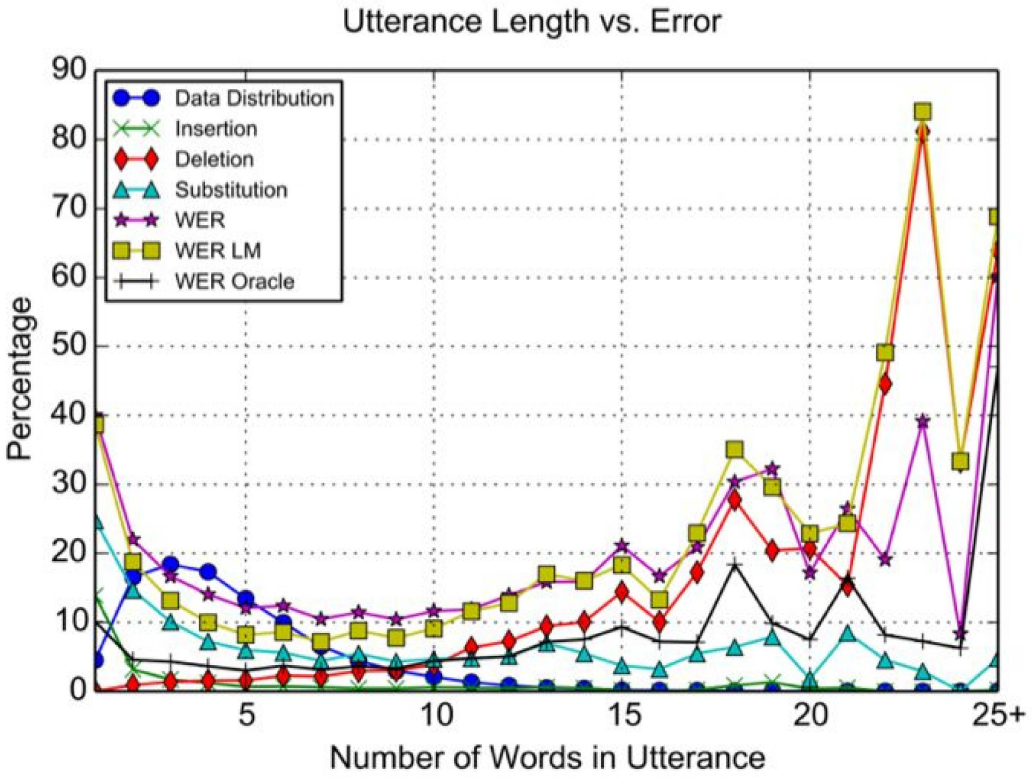
William Chan, Navdeep Jaitly, Quoc Le, Oriol Vinyals. *Listen Attend and Spell*. ICASSP 2015.

Limitations of LAS (seq2seq)

- Not an online model - input must all be received before transcripts can be produced
- Attention is a computational bottleneck since every output token pays attention to every input time step
- Length of input has a big impact on accuracy

Impact of Length

- Naive decoding results in early termination of outputs

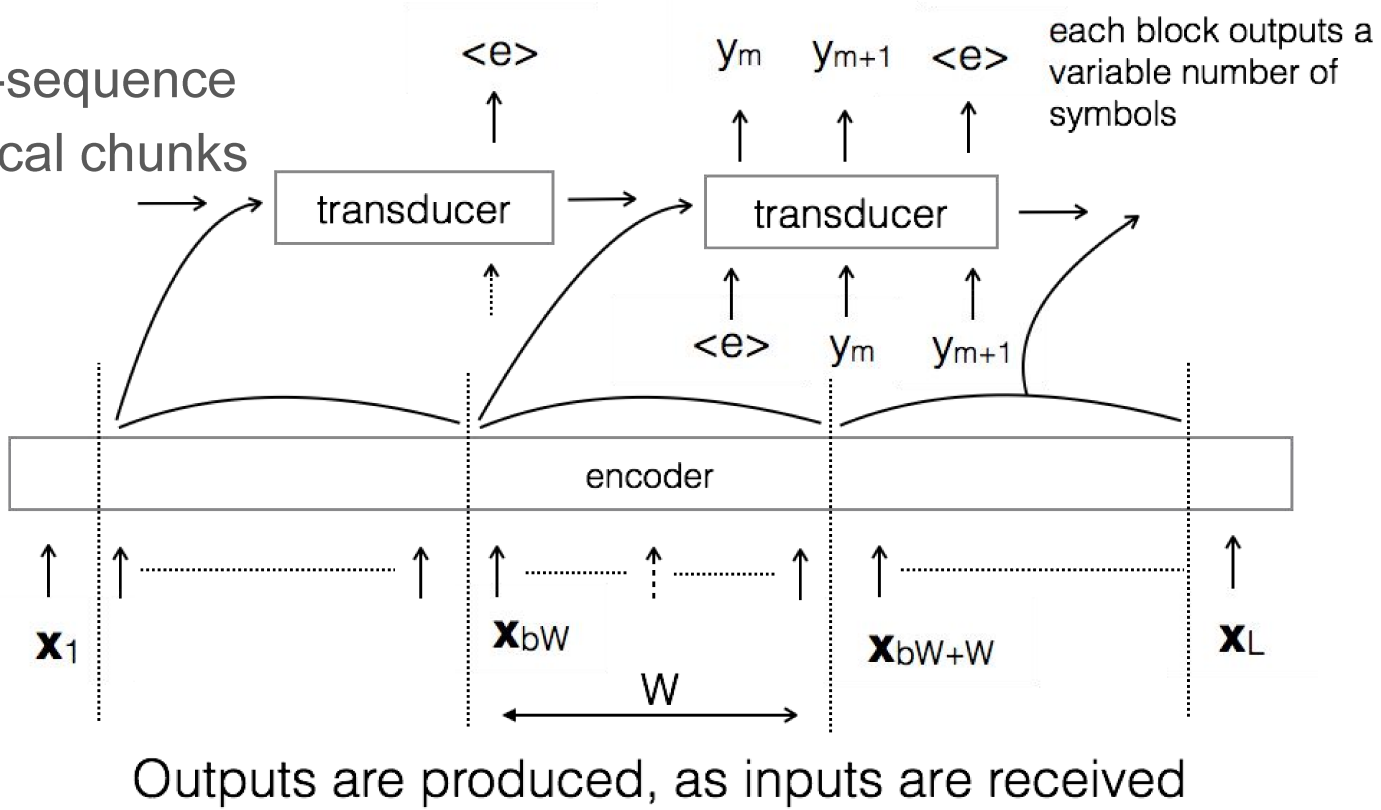


Online Sequence to Sequence Models

- Overcome limitations of the sequence to sequence models
  - No need to wait for the entire input sequence to arrive
  - Attention over the entire sequence is an overkill (attention is very locally peaked)
- Produce outputs as inputs arrive
  - Has to solve the following problem:
    - When has enough information arrived that the model is confident enough to output symbols

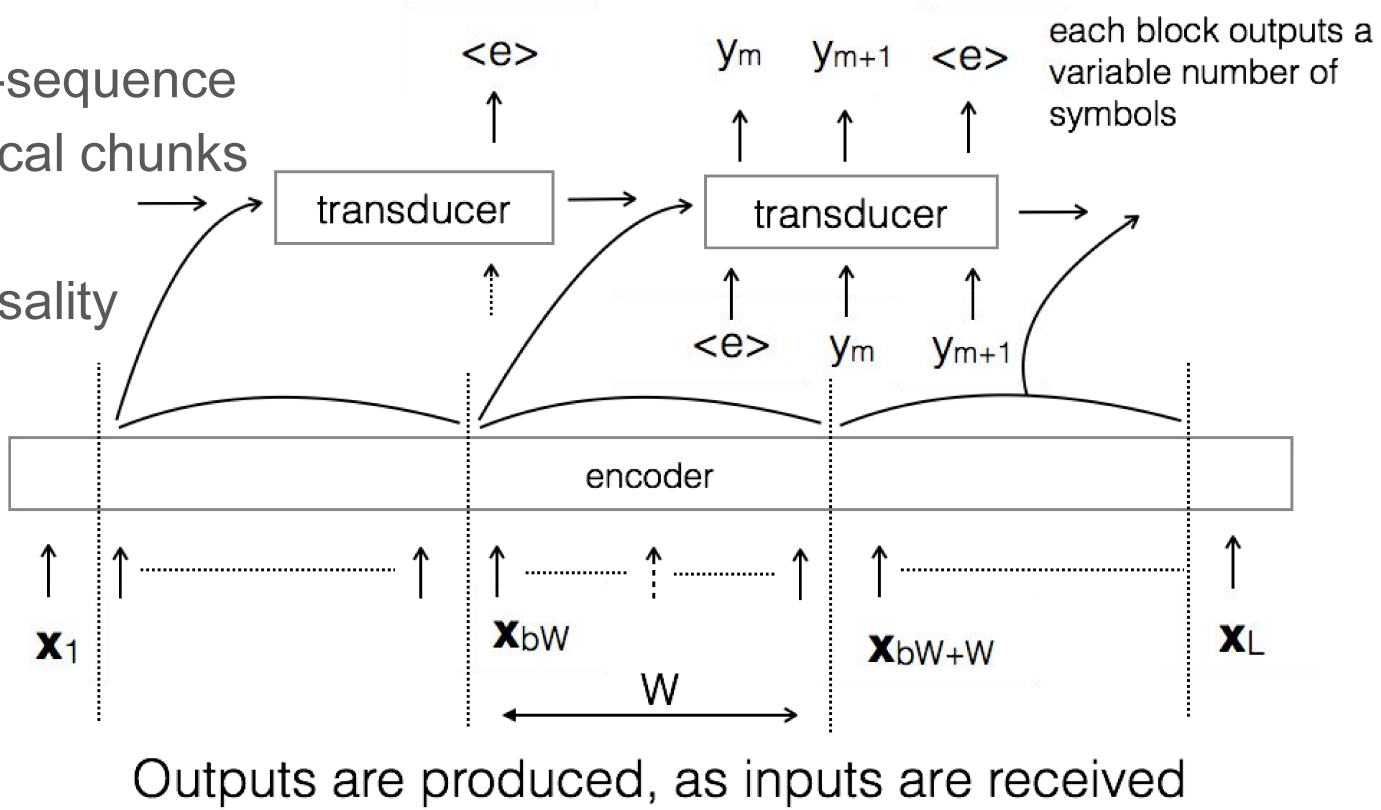
A Neural Transducer

- sequence-to-sequence models on local chunks of data



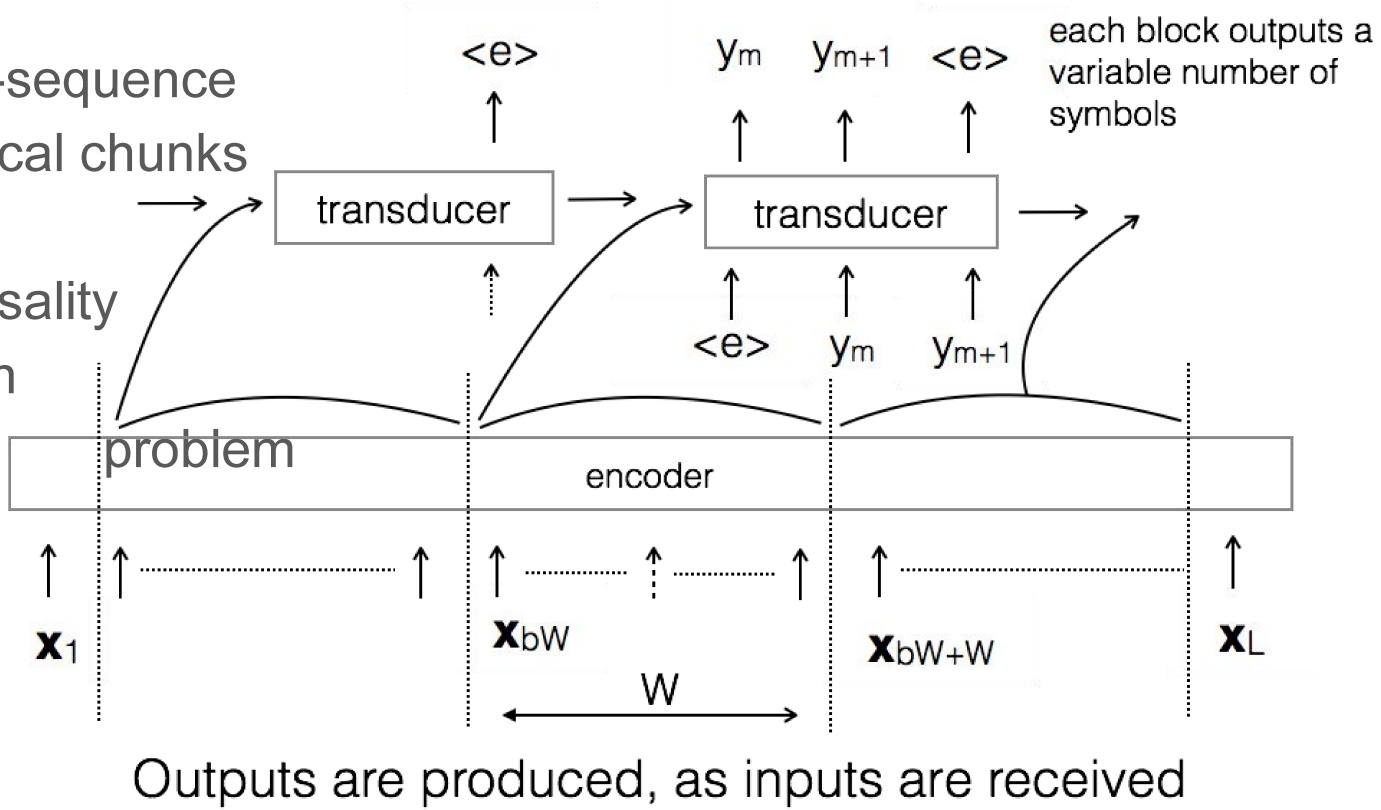
A Neural Transducer

- sequence-to-sequence models on local chunks of data
- Maintain causality



A Neural Transducer

- sequence-to-sequence models on local chunks of data
- Maintain causality
- Introduces an alignment problem



A Neural Transducer

- The output y can be produced from a combinatorial number of ways from the data

$$p(y_{1\ldots S}|x) = \sum_{\tilde{y}_{1\ldots(S+B)} \in \mathcal{Y}} p(\tilde{y}_{1\ldots S+B}|x)$$

  - $\tilde{y}_{1..(S+B)}$ includes <e> and vocabulary
  - $y_{1..S}$ is the actual output sequence
- Inference is done by using beam search to find highest probability output sequence for an input.

$$y_{1\ldots S} = \arg \max_{\tilde{y}_{1\ldots S'}, e_{1\ldots N}} \sum_{b=1}^{N} \log p(\tilde{y}_{(e_{b-1}+1)\ldots e_b}|x_{1\ldots bW}, \tilde{y}_{1\ldots e_{b-1}})$$

A Neural Transducer

- The output y can be produced from a combinatorial number of ways from the data

$$p(y_{1\ldots S}|x) = \sum_{\tilde{y}_{1\ldots(S+B)} \in \mathcal{Y}} p(\tilde{y}_{1\ldots S+B}|x)$$

  - $\tilde{y}_{1..(S+B)}$ includes <e> and vocabulary
  - $y_{1..S}$ is the actual output sequence
- Inference is done by using beam search to find highest probability output sequence for an input.

$$y_{1\ldots S} = \arg \max_{\tilde{y}_{1\ldots S'}, e_{1\ldots N}} \sum_{b=1}^{N} \log p(\tilde{y}_{(e_{b-1}+1)\ldots e_b}|x_{1\ldots bW}, \tilde{y}_{1\ldots e_{b-1}})$$

A Neural Transducer - Training

- Correct gradient of log likehood:

$$\sum_{\tilde{y} \in \mathcal{Y}} p\left(\tilde{y}_{1\cdots(S+B))} | \mathbf{x}_{1\cdots L}, y_{1\cdots S}\right) \frac{d}{d\theta} \log p\left(\tilde{y}_{1\cdots(S+B)} | \mathbf{x}_{1\cdots L}\right)$$

- Viterbi-like training works well in practice:

$$\frac{d}{d\theta} \log p\left(\tilde{y}_{1\cdots(S+B)} | \mathbf{x}_{1\cdots L}\right)$$

$$\tilde{y}_{1\cdots(S+B)} = \arg \max_{\hat{y}_{1\cdots(S+B)}} p\left(\hat{y}_{1\cdots(S+B))} | \mathbf{x}_{1\cdots L}, y_{1\cdots S}\right)$$

(forced alignment with a DNN-HMM system works well too!)

A Neural Transducer - Finding best path

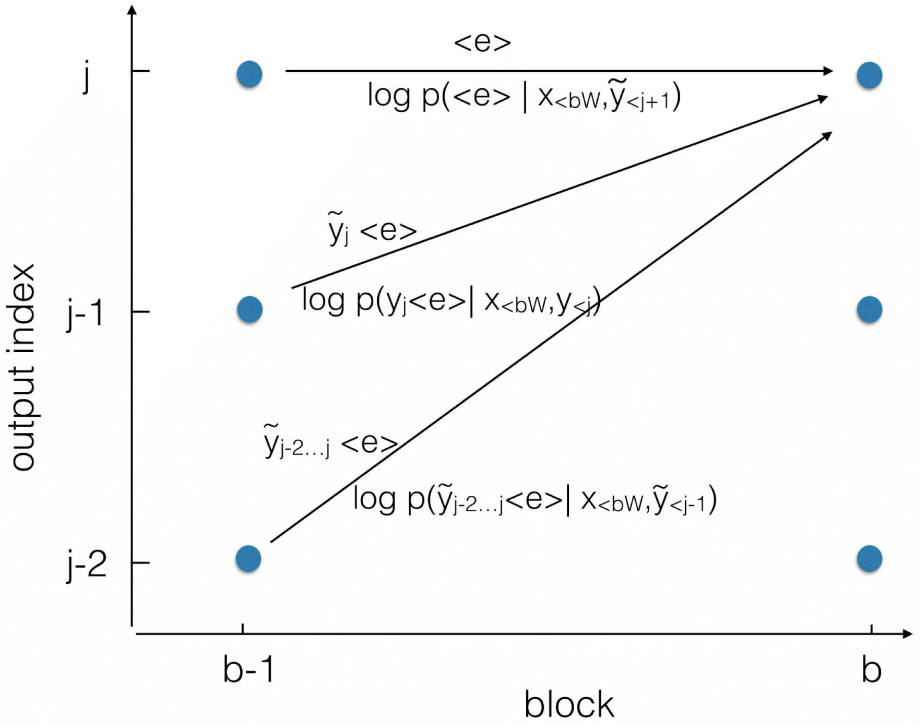
- Finding best path is tricky -- beam search fails easily

$$\tilde{y}_{1\cdots(S+B)} = \arg \max_{\hat{y}_{1\cdots(S+B)}} p\left(\hat{y}_{1\cdots(S+B)}) | \mathbf{x}_{1\cdots L}, y_{1\cdots S}\right)$$

- Approximate Dynamic programming works well

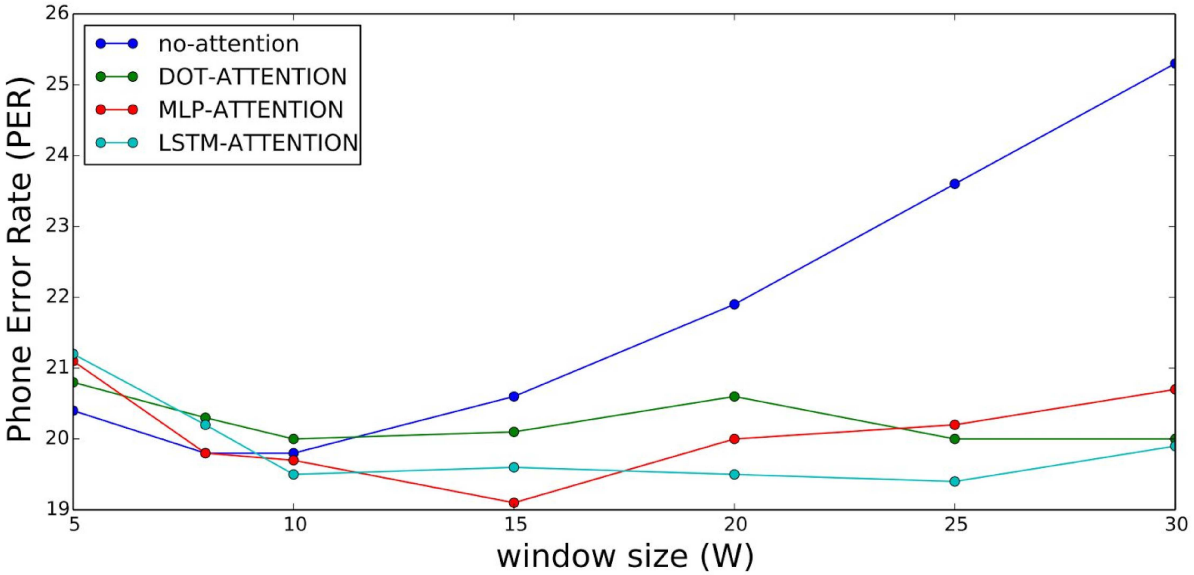
A Neural Transducer - Dynamic programming

- Approximate Dynamic programming -- finding best alignment



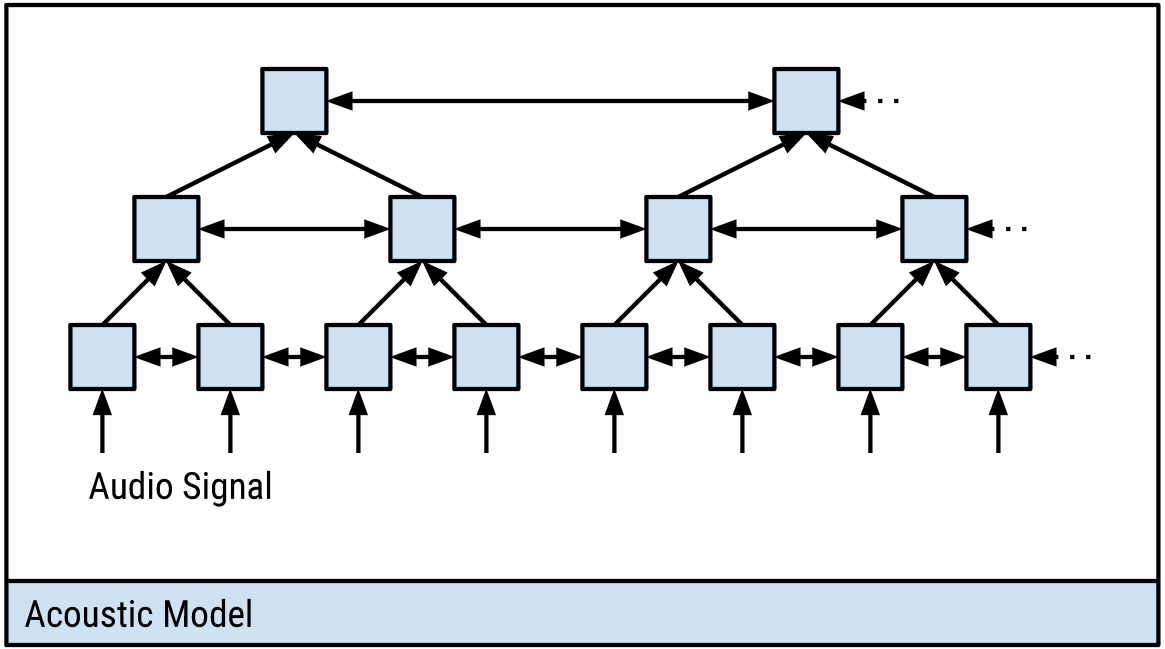
A Neural Transducer - Results

# of layers in encoder / transducer	1	2	3	4
2		19.2	18.9	18.8
3		18.5	**18.2**	19.4



Very Deep Convolutional Encoders for LAS

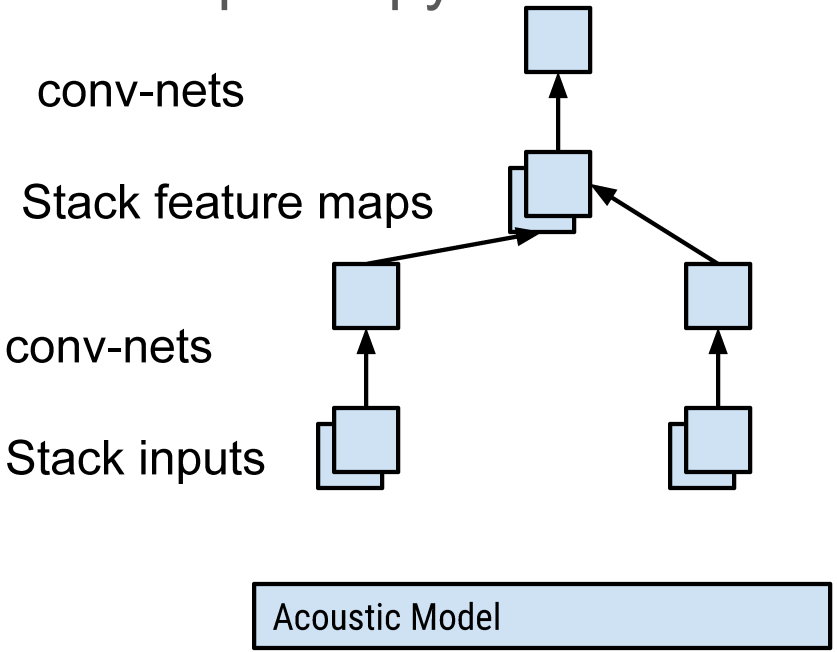
- Pyramidal RNN reduces the time resolution of the inputs

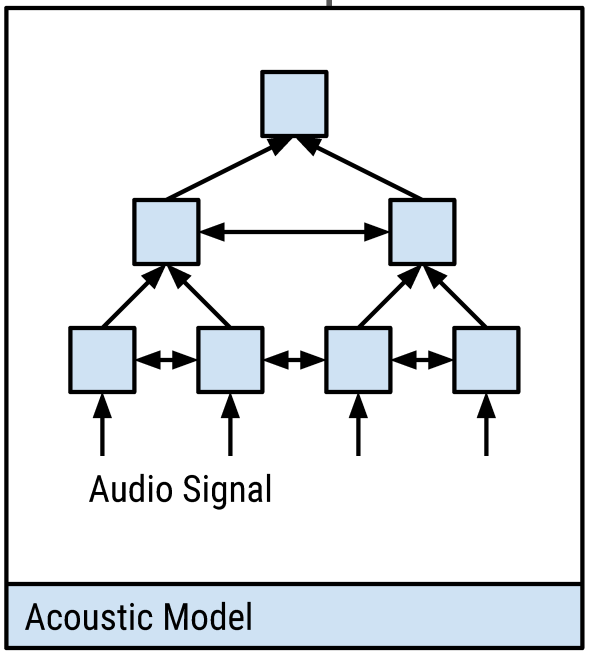


Yu Zhang, William Chan, Navdeep Jaitly. *Very Deep Convolutional Networks for End-to-End Speech Recognition*. https://arxiv.org/abs/1610.03022.

Very Deep Convolutional Encoders for LAS

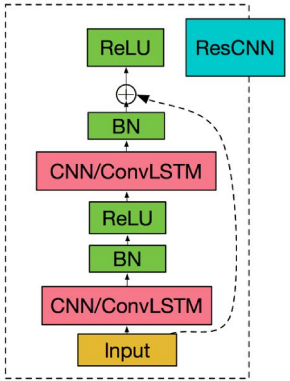
- Replace pyramid with convolutions -- adds depth



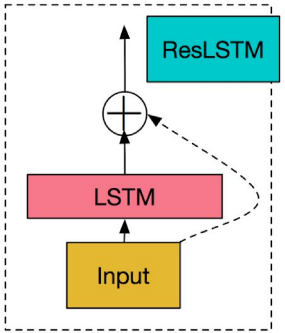


Yu Zhang, William Chan, Navdeep Jaitly. *Very Deep Convolutional Networks for End-to-End Speech Recognition*. https://arxiv.org/abs/1610.03022.

Very Deep Convolutional Encoders for LAS (WSJ test_eval ‘92)



(a) ResCNN block



(b) ResLSTM block

Model	WER
L × 3	14.76
ConvLSTM × 3	24.23
(C (3×3)) × 2 + ResCNN × 4 + NiN	11.30
(C (3×3)) × 2 + ResConvLSTM (3 × 1) × 4 + NiN	10.53

Table 3: Performance of models with convolutional LSTM layers. The NiN block equals (L + C (1x1) + B + R) × 2 + L).

* *No language models or dictionaries*

Yu Zhang, William Chan, Navdeep Jaitly. *Very Deep Convolutional Networks for End-to-End Speech Recognition*. https://arxiv.org/abs/1610.03022.

Choosing the correct output targets

- Each output sequence is represented by an unique fixed representation:
  - “HELLO CHARMANDER”
    - CHARMANDER is a rare pokemon
- Word:
  - [“HELLO”, “CHARMANDER”]
    - “CHARMANDER” is a OOV/rare word -- large softmax? Overfit?
- Characters:
  - [“H”, “E”, “L”, “L”, “O”, “ “, “C”, “H”, “A”, “R”, “M”, “A”, “N”, “D”, “E”, “R”]
    - Long decoding length
    - Listen, Attend and Spell (Chan et al., 2016; Bahdanau et al., 2016)
- Word + Characters:
  - [“HELLO”, “ “, “C”, “H”, “A”, “R”, “M”, “A”, “N”, “D”, “E”, “R”]
    - MT: Words vocab, use characters for rare/OOV words (Thang et al., 2016)

Choosing the correct output targets - Word Pieces

- Word Pieces from N-grams of Characters
  - Count n-grams from text, take top 256/512/1024 word pieces
  - Possibly better units to model speech? Closer to phonemes.
    - “th”, “he”, “in”, “ing”, “tion”, “est”
- Multiple Decompositions per Sequence
  - “HELLO”
    - [“H”, “E”, “L”, “L”, “O”]
    - [“HE”, “L”, “L”, “O”]
    - [“HE”, “LLO”]
    - [“HELL”, “O”]
    - … combinatorial # of choices ...

William Chan, Yu Zhang, Quoc Le, Navdeep Jaitly. *Latent Sequence Decompositions*. https://arxiv.org/abs/1610.03035.

Choosing the correct output targets - Word Pieces

- Which choice of tokens should we decompose our target sequences with ?
  - Greedy left-to-right ? CAT SITS -> CA, T, SI, T, S
  - Compression ? CAT SITS -> C, AT, SI, T, S
  - Expert Heuristics?

Word Piece Dictionary: A-Z, CA, AT, SI

Target Sentence: CAT SITS

William Chan, Yu Zhang, Quoc Le, Navdeep Jaitly. *Latent Sequence Decompositions*. https://arxiv.org/abs/1610.03035.

Latent Sequence Decompositions

- Marginalize all possible paths:

$$\begin{aligned}\log p(\mathbf{y}^*|\mathbf{x};\theta) &= \log \sum_{\mathbf{z}} p(\mathbf{y}^*, \mathbf{z}|\mathbf{x};\theta) \\ &= \log \sum_{\mathbf{z}} p(\mathbf{y}^*|\mathbf{z},\mathbf{x}) p(\mathbf{z}|\mathbf{x};\theta) \\ &= \log \sum_{\mathbf{z}} p(\mathbf{y}^*|\mathbf{z}) p(\mathbf{z}|\mathbf{x};\theta)\end{aligned}$$

- MLE gradient estimator:

$$= \mathbb{E}_{z \sim p(z|x;\theta), z \in \{z' : p(y^*|z')=1\}} \left[\nabla \log p(z|x;\theta)\right]$$

William Chan, Yu Zhang, Quoc Le, Navdeep Jaitly. *Latent Sequence Decompositions*. https://arxiv.org/abs/1610.03035.

Latent Sequence Decompositions (WSJ test_eval92)*

Model	WER	WER (greedy)
CTC+character (Graves et al., 2014)	27.3	-
seq2seq+character	14.7	-
2-gram word piece	13.2	15.3
3-gram word piece	12.9	15.7
4-gram word piece	13.1	15.3
5-gram word piece	13.2	15.1

* *No language models or dictionaries*

Qualitative Results

shamrock's pretax profit from the sale was one hundred twenty five million dollars a spokeswoman said

character:
c|h|a|m|r|o|c|k|'|s| |p|r|e|t|a|x| |p|r|o|f|i|t| |f|r|o|m| |t|h|e| |s|a|l|e| |w|a|s| |o|n|e| |h|u|n|d|r|e|d| |t|w|e|n|t|y| |f|i|v|e| |m|i|l|l|i|o|n| |d|o|l|l|a|r|s| |a| |s|p|o|k|e|s|w|o|m|a|n| |s|a|i|d -1.373868

4-gram
sh|a|m|ro|c|k|'s| |pre|ta|x| |**pro**|fi|t| |**fro**|m| |t|h|e| |sa|l|e| |was| |on|e| |hu|n|dr|e|d| |t|we|nt|y| |**fiv**|e| |**mil**|**lio**|n| |**doll**|a|r|s| |a| |sp|ok|e|s|wo|ma|n| |said -28.111485

- “shamrock” is in-vocab, “shamrock’s” is OOV.
- Bolded parts to highlight the 3/4-grams

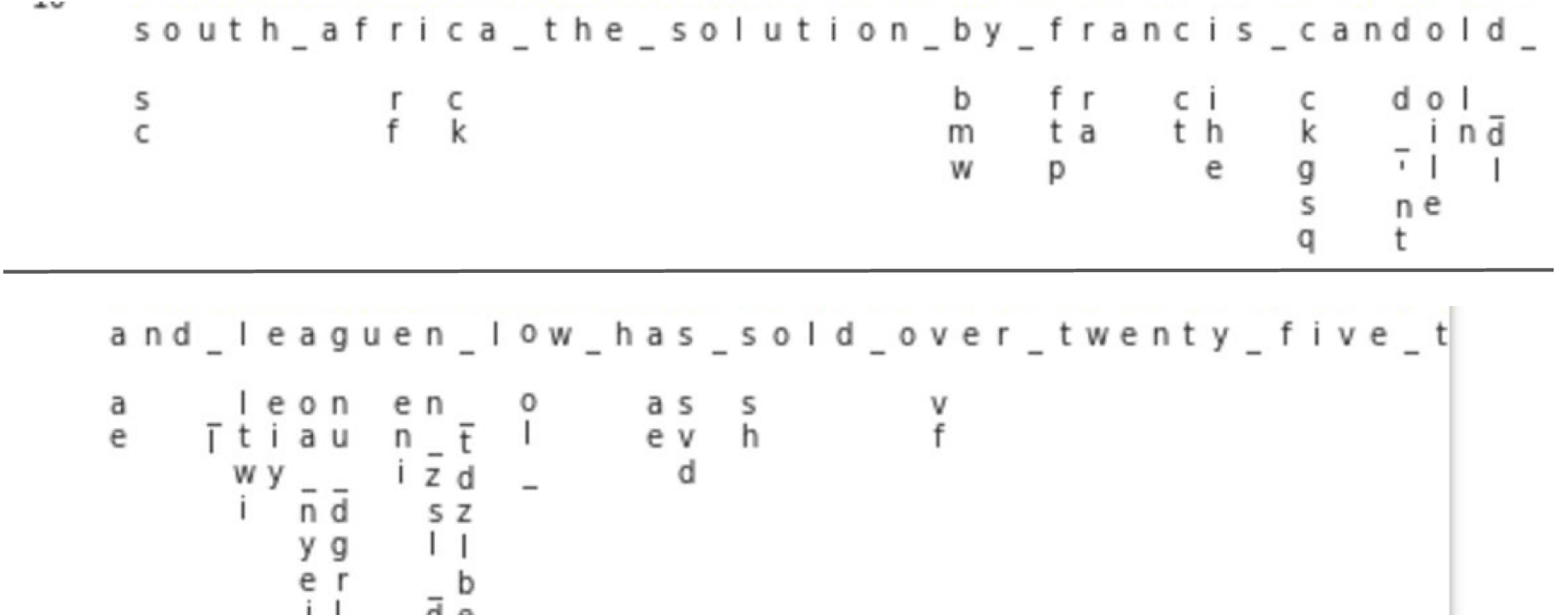
N-gram usage for 4-gram model

- Token % is the % of tokens used to model word
- Character % is the corresponding number of chars.

N	MLE		Maximum Extension	
	Token %	Character %	Token %	Character %
1	62.45	42.10	18.03	8.24
2	28.15	37.96	51.02	46.62
3	8.00	16.19	25.04	34.32
4	1.39	3.75	5.92	10.82
WER	13.1		15.3	

Model Shortcomings

- More ambiguity at start of words



Overconfidence at word boundaries

- Most discrimination starts at the beginnings of words (or throughout the word for confusing words such as names)

- Wrong overconfidence at word boundaries can cause issues in searching, and language model blending

Jan Chorowski, Navdeep Jaitly. *Towards better decoding and language model integration in sequence to sequence models*. https://arxiv.org/abs/1612.02695.

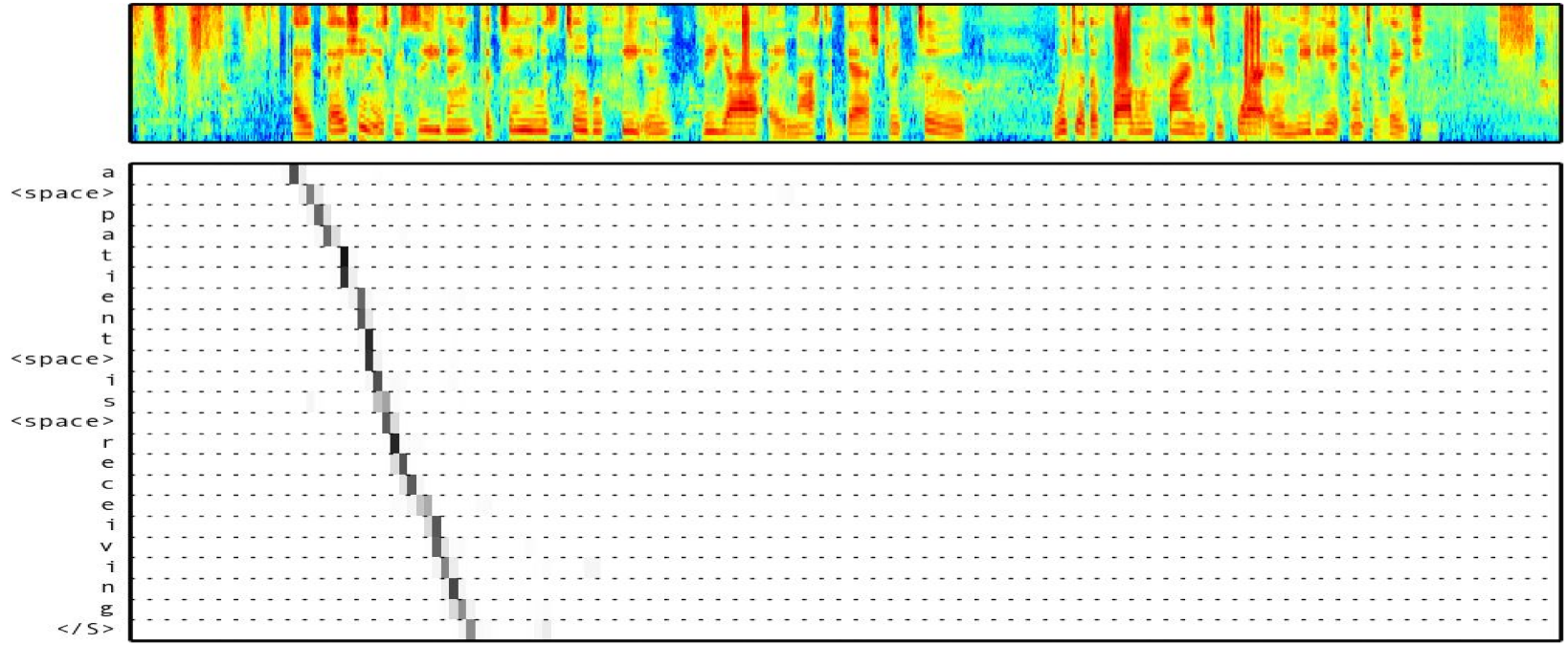
Overconfidence at word boundaries

- Entropy Regularization of target distribution at each step counters overconfidence

Model	Parameters	dev93	eval92
CTC [3]	26.5M	-	27.3
seq2seq [12]	5.7M	-	18.6
seq2seq [24]	5.9M	-	12.9
seq2seq [22]	-	-	10.5
Baseline	6.6M	17.9	14.2
Unigram LS	6.6M	**13.7**	**10.6**
Temporal LS	6.6M	14.1	10.7

Jan Chorowski, Navdeep Jaitly. *Towards better decoding and language model integration in sequence to sequence models*. https://arxiv.org/abs/1612.02695.

Lack of generative penalty



Decoding of long utterances can terminate outputs quickly since the data does not need to be explained

Lack of generative penalty

Discriminative model chooses to terminate sequence early, or even skip parts of input

Table 1: *Example of model failure on validation '4k0c030n'*

Transcript	LM cost $\log p(y)$	Model cost $\log p(y\|x)$
"chase is nigeria's registrar and the society is an independent organization hired to count votes"	-108.5	-34.5
"in the society is an independent organization hired to count votes"	-64.6	-19.9
"chase is nigeria's registrar"	-40.6	-31.2
"chase's nature is register"	-37.8	-20.3
""	-3.5	-12.5

Jan Chorowski, Navdeep Jaitly. *Towards better decoding and language model integration in sequence to sequence models*. https://arxiv.org/abs/1612.02695.

Lack of generative penalty

- Adding coverage reward during decoding resolves this problem

$$\text{coverage} = \sum_j \left[\sum_i \boldsymbol{\alpha}_{ij} > \tau \right]$$

Jan Chorowski, Navdeep Jaitly. *Towards better decoding and language model integration in sequence to sequence models*. https://arxiv.org/abs/1612.02695.

Lack of generative penalty

- Adding coverage penalty during decoding resolves this problem

$$\text{coverage} = \sum_j \left[\sum_i \alpha_{ij} > \tau \right]$$

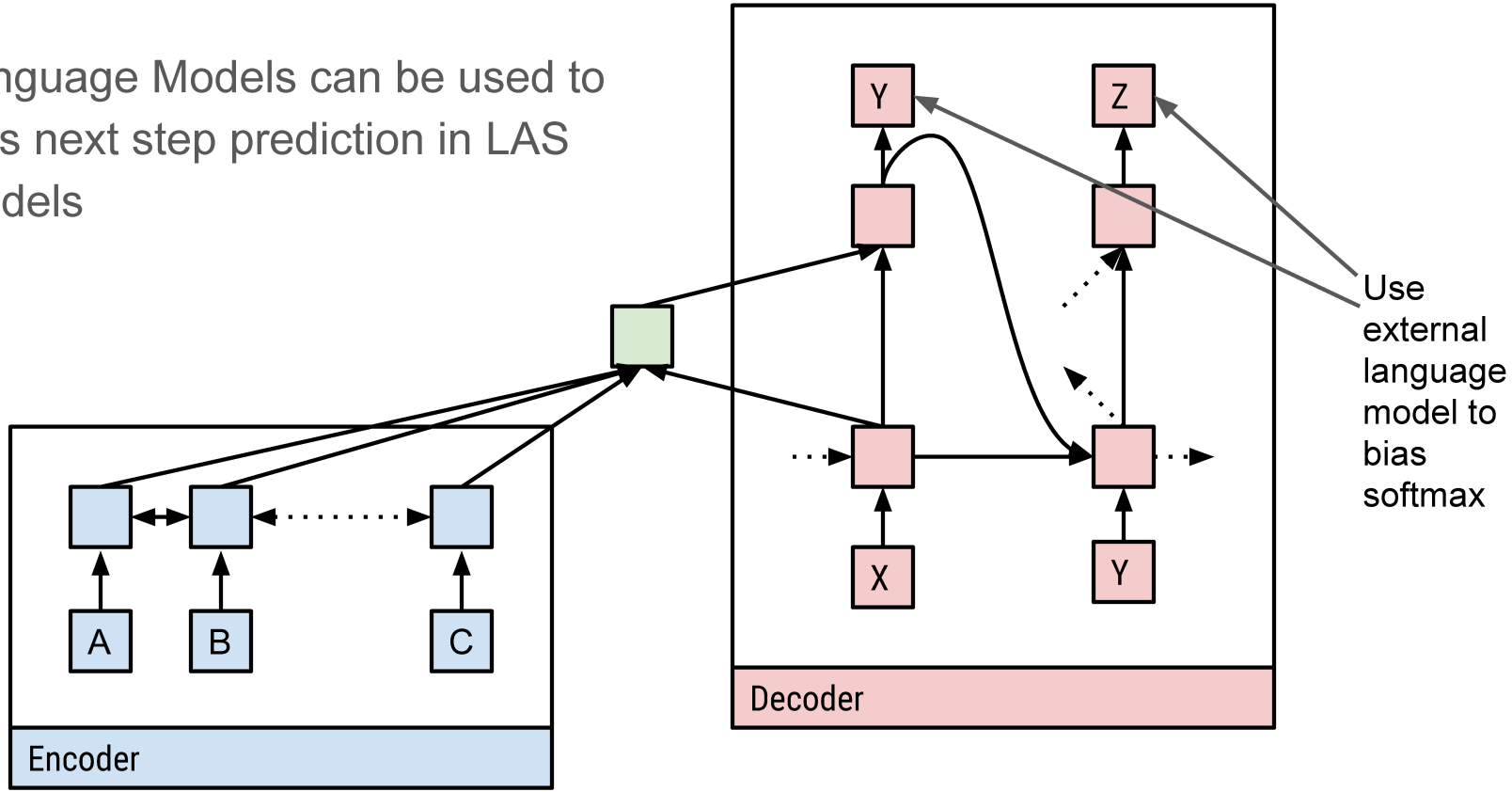
Model	dev93	eval92
seq2seq [12]	-	9.3
CTC [3]	-	8.2
CTC [6]	-	7.3
Baseline + Cov	12.6	8.9
Unigram LS + Cov.	9.9	7.0
Temporal LS + Cov.	**9.7**	**6.7**

Uses trigram language model

Jan Chorowski, Navdeep Jaitly. *Towards better decoding and language model integration in sequence to sequence models*. https://arxiv.org/abs/1612.02695.

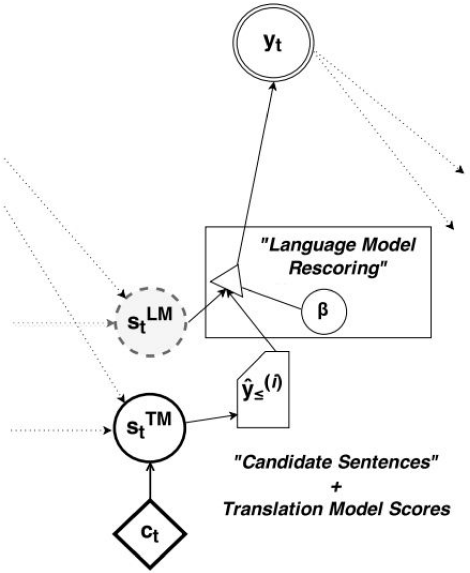
Better Language Model Blending

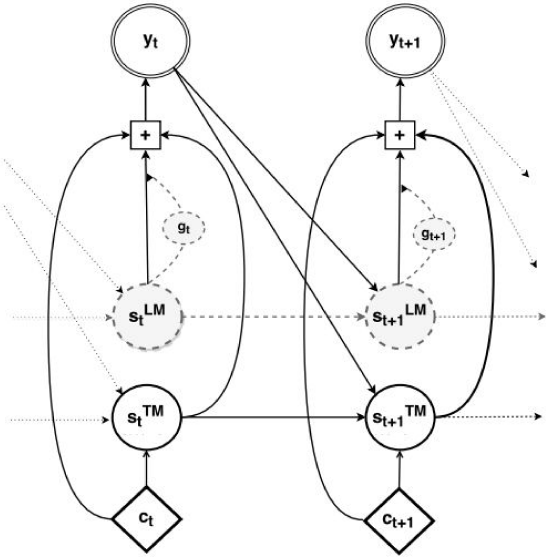
- Language Models can be used to bias next step prediction in LAS models



Better Language Model Blending

- Shallow and Deep Fusion Models in Machine Translation





TM - translation model
LM - language model

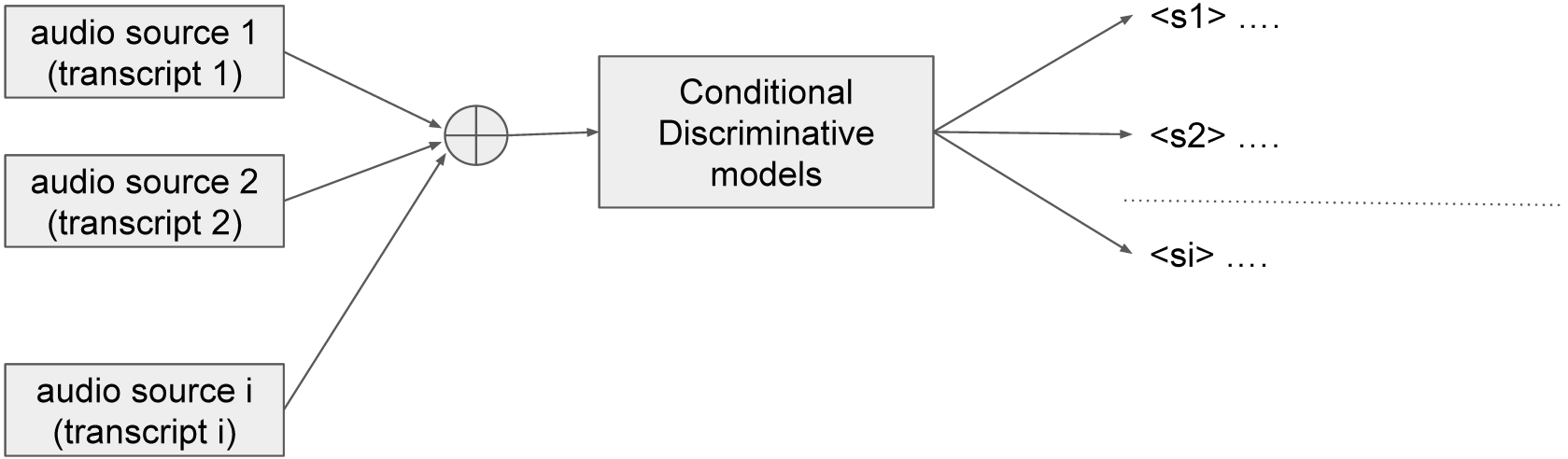
Caglar Gulchere, et. al. *On Using Monolingual Corpora in Neural Machine Translation*. https://arxiv.org/abs/1503.03535.

Better Sequence Training of Models

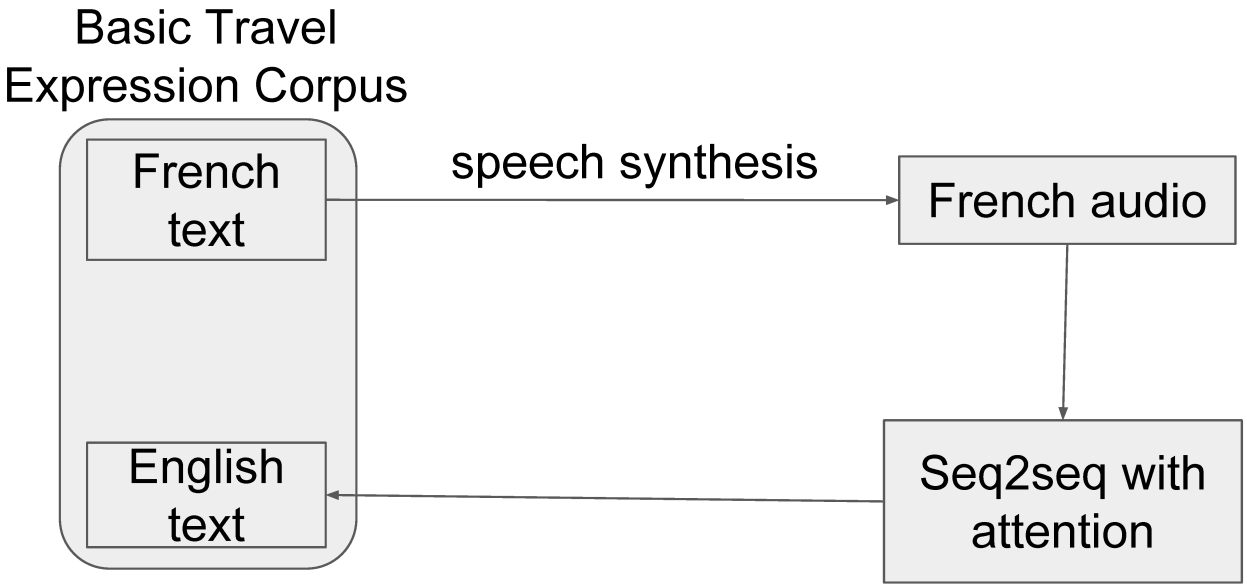
- Training focusses on next step prediction without considering impact of prediction on long term horizon
- Optimizing with long sequences in mind should better capture long range structure
  - Scheduled Sampling (Bengio et. al. 2015)
  - WER optimization with Reinforcement Learning (Ranzato et. al. 2015)
  - Sequence-to-sequence as beam search optimization (Wiseman & Rush, 2016)

Opportunities - Multispeaker / Multichannel setup

- Multimodal model allows for distinct outputs corresponding to the same input

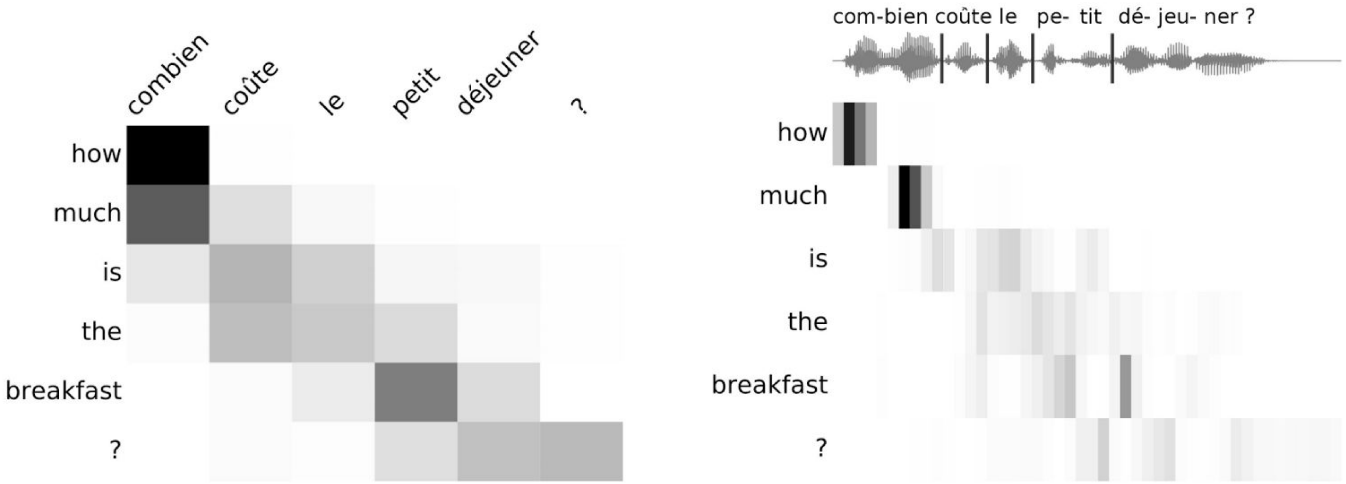


Opportunities - Direct Translation



Alexandre Berard, Olivier Pietquin, Christophe Servan, Laurent Besacier. *Listen and Translate: A Proof of Concept for End-to-End Speech-to-Text Translation*. https://arxiv.org/abs/1612.01744.

Opportunities - Direct Translation



(a) Machine translation alignment

(b) Speech translation alignment

Alexandre Berard, Olivier Pietquin, Christophe Servan, Laurent Besacier. *Listen and Translate: A Proof of Concept for End-to-End Speech-to-Text Translation*. https://arxiv.org/abs/1612.01744.

Opportunities - Direct Translation

Corpus	Speaker	BLEU score				
		Greedy	+BeamSearch	+LM	+Ensemble	Baseline
dev	*Agnes*	30.1	32.3	33.5	40.2	43.7
test1	*Agnes*	29.2	31.9	32.7	**38.6**	**42.1**
test2	*Agnes*	29.0	30.6	31.0	**37.1**	**39.6**
test1 (7 refs)	*Agnes*	37.6	40.1	41.8	**48.8**	**52.7**
	Michel	40.6	43.1	44.4	**50.4**	**51.9**
test2 (7 refs)	*Agnes*	33.6	35.6	36.7	**43.1**	**46.7**
	Michel	37.3	39.6	41.0	**46.7**	**46.4**
train.1000	Michel	53.8	60.8	61.8	82.6	60.5

Alexandre Berard, Olivier Pietquin, Christophe Servan, Laurent Besacier. *Listen and Translate: A Proof of Concept for End-to-End Speech-to-Text Translation*. https://arxiv.org/abs/1612.01744.

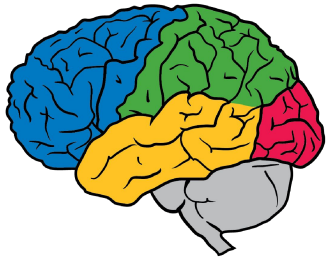
Acknowledgements

LAS

- **William Chan***
- ***Yu Zhang***
- Jan Chorowski
- Quoc Le
- Tara Sainath

Online Sequence to Sequence

- **Yuping Luo**
- Chung-Cheng Chui
- Dieterich Lawson
- George Tucker
- Ilya Sutskever



Google Brain Team -- Ashish Agarwal, Zhifeng Chen, Xin Pan, Jon Shlens