End-to-End Attention-Based Large Vocabulary Speech Recognition

Dzmitry Bahdanau Jan Chorowski Dmitriy Serdyuk Philemon Brakel Yoshua Bengio

University of Montreal (MILA)

2016 IEEE International Conference on Acoustics, Speech and Signal Processing

Motivation

Problem Statement

- Current SOTA contain separate modules for acoustic and language modelling, sequence decoding etc.
- Deep networks are popular acoustic models, but are hybrid systems (GMM + HMM) which require a two-stage training process.
- Acoustic model is not directly trained to minimize the final objective of interest.

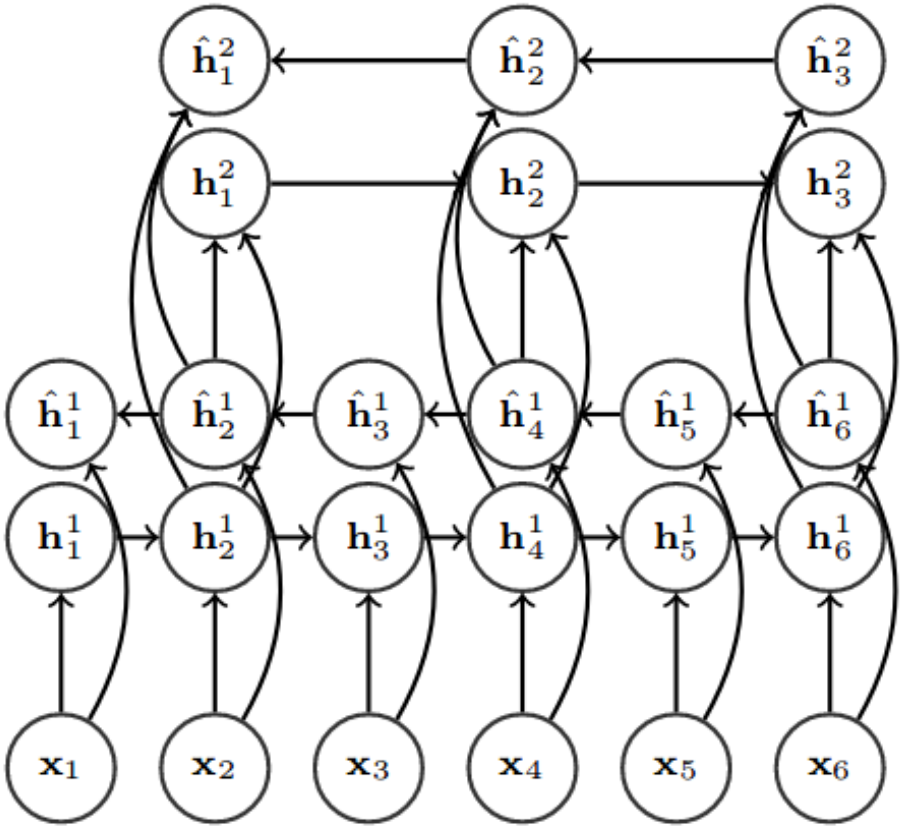
Motivation

Contributions

- Train end-to-end models by replacing HMMs with Attention based Recurrent Sequence Generators (ARSG)
- Long sequences can be modeled by limiting the area of attention to a range of most promising locations.
- A recurrent architecture that reduces source sequence length by pooling frames neighbouring in time.
- Combine character level ARSG and n-gram model using Weighted Finite State Transducer framework

Attention-Based Recurrent Sequence Generators

Encoder-Decoder Architecture

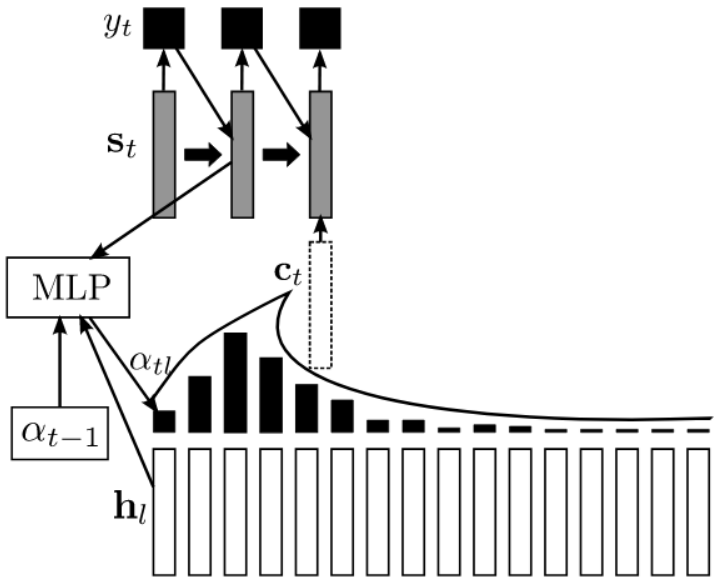


Architecture of Encoder: Bi-Directional RNN with Pooling over Time

Attention-Based Recurrent Sequence Generators

Encoder-Decoder Architecture

$$p(y_1, y_2, \ldots, y_T | h_1, h_2, \ldots, h_l)$$



Architecture of Decoder

$$\mathbf{F} = \mathbf{Q} * \boldsymbol{\alpha}_{t-1}$$

$$e_{tl} = \mathbf{w}^\top \tanh(\mathbf{W}\mathbf{s}_{t-1} + \mathbf{V}\mathbf{h}_l + \mathbf{U}\mathbf{f}_l + \mathbf{b})$$

$$\alpha_{tl} = \exp(e_{tl}) \Big/ \sum_{l=1}^{L} \exp(e_{tl}) \, .$$

Attention-Based Recurrent Sequence Generators

Windowing

- What is the complexity of the training procedure? (for attention weights)

Attention-Based Recurrent Sequence Generators

Windowing

- What is the complexity of the training procedure? (for attention weights)
- Complexity: O(LT)
- Is it possible to achieve a lower time complexity?

Attention-Based Recurrent Sequence Generators

Windowing

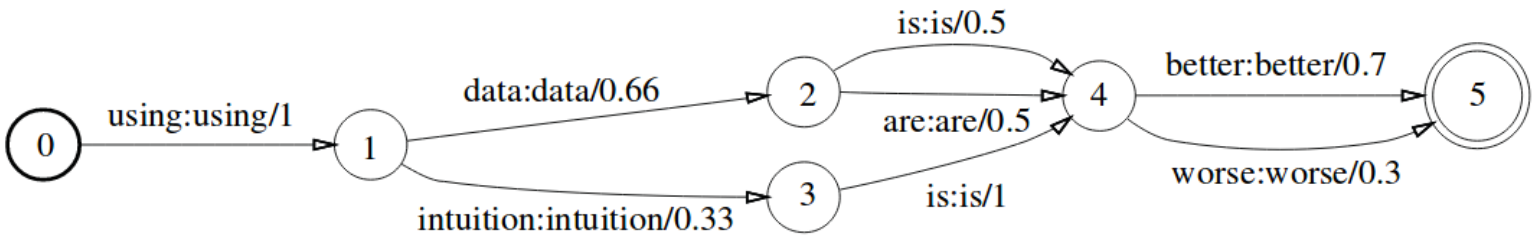
- What is the complexity of the training procedure? (for attention weights)
- Complexity: O(LT)
- Is it possible to achieve a lower time complexity?
- Use Windowing: attention mechanism considers only positions within the context window

Integration with Language Model

- ARSG implicitly learns how an output symbol depends on previous ones.
- Transcripts of training sets not sufficient to learn a high-quality language model.
- Problem: Standard speech recognition tasks use word-base language models whereas ARSG is a character based model.

Integration with Language Model

- Use Weighted Finite State Transducer to build character level model



Sample WFST

Integration with Language Model

- Let L be the lexicon FST and G be the language model.

$$newFST = min(det(L \circ G))$$

$$L = -logp_{ED}(y|x) - \beta logp_{LM}(y) - \gamma T$$

- To minimize L, approximate the value of y using beam search.

Integration with Language Model

Beam Search

- Choose k most likely candidates at each step
- Expand on those candidates. Repeat.
- To avoid underflow, use natural logarithm
- Terminate when certain conditions are met (EOS, threshold etc)

Results

Data

- Wall Street Journal Corpus
- 81 hour long with a set of 37K sequences
- Feature Representation: MFCCs

Results

Model	CER	WER
Encoder-Decoder	6.4	18.6
Encoder-Decoder + bigram LM	5.3	11.7
Encoder-Decoder + trigram LM	4.8	10.8
Encoder-Decoder + extended trigram LM	3.9	9.3
Graves and Jaitly (2014)		
CTC	9.2	30.1
CTC, expected transcription loss	8.4	27.3
Hannun et al. (2014)		
CTC	10.0	35.8
CTC + bigram LM	5.7	14.1
Miao et al. (2015),		
CTC for phonemes + lexicon	-	26.9
CTC for phonemes + trigram LM	-	7.3
CTC + trigram LM	-	9.0

Discussion Questions

- Why is ARSG performing better than CTC, inspite of not using a language model?
- Would a small dataset (as used in the paper) affect ARSG's performance?