CPSC 503 - Intro to E2E ASR

Peter Sullivan - April 24th 2020

Lecture Overview

- Intro to ASR
- Features in ASR
- Traditional Approaches
- Overview of E2E-ASR (examples of lecture slides)
- CTC
- Decoding
- Improvements to CTC ASR
- Future Work

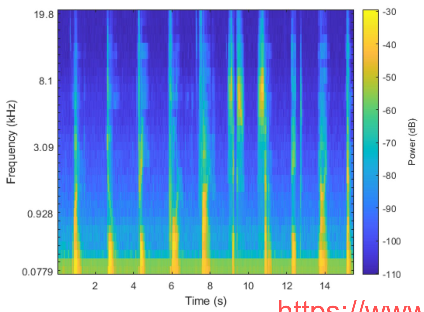
Introduction to ASR

End-to-End Automatic Speech Recognition

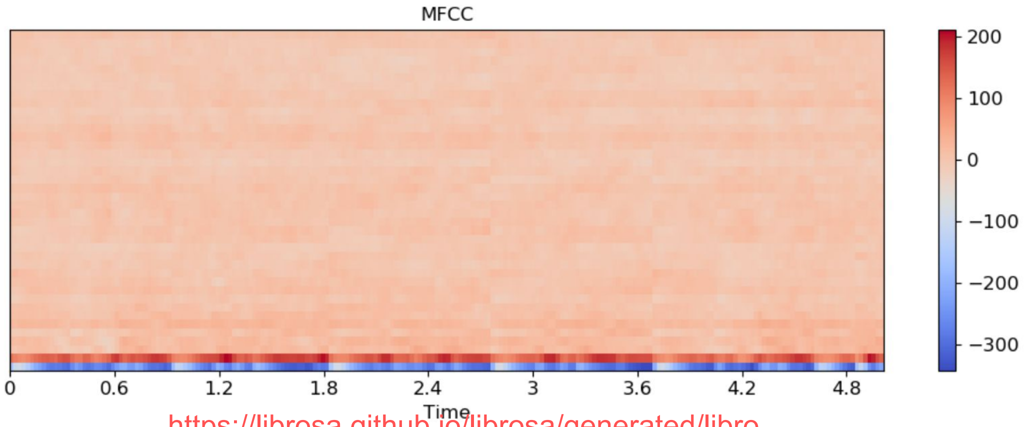
- You probably use it already!
- Google, Amazon, Apple have pioneered applications
- Integrates with many other parts of NLP
    - Question Answering
    - Summarization
    - State Detection / Emotion Detection

Features in ASR

- Mel Spectrogram
  - Mel scale spectrogram to capture more
- MFCC
  - Sound transform to better emulate human hearing
- Raw Wave files
  - These work too!
  - wav2vec uses these!



https://www.mathworks.com/help/audio/ref/melspectrogram.html



https://librosa.github.io/librosa/generated/librosa.feature.mfcc.html

Overview of Traditional ASR

Traditional Speech Recognition Model:

- Acoustic Model: Hidden Markov Model / Gaussian Mixture Model based
    - DNN sometimes used instead of GMM (Training implications)
- Language Model: n-gram
- Decoding: Beam or Viterbi
- Annotation/Alignment
    - Human Error/Need high skill

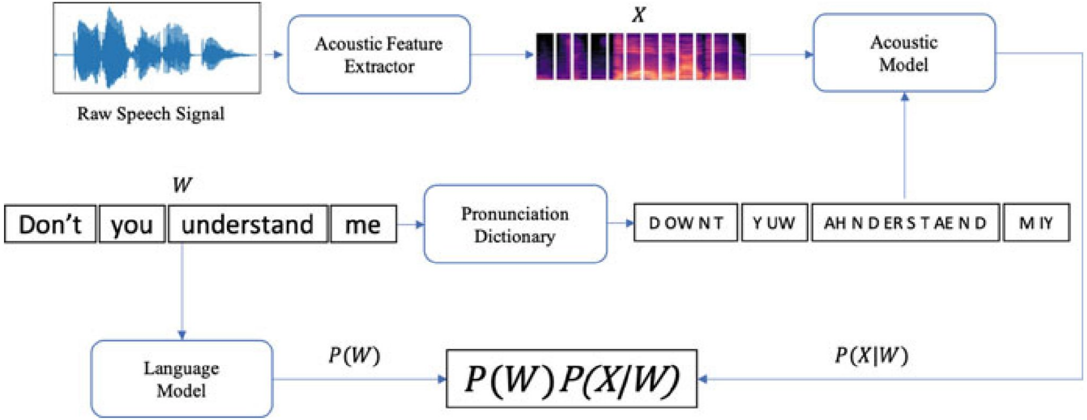


Image: Kamath, U., Liu, J., & Whitaker, J. (2019). Deep learning for nlp and speech recognition. Springer International Publishing.

E2E ASR

Can we avoid the downside in annotating/aligning with a model trained together?

- Neural Model (CNN-RNN)
- Connectionist Temporal Classification (CTC) or Attention-Based approaches
- Can improve with addition of LM and decoding
- Needs lots of data

Typical model family:

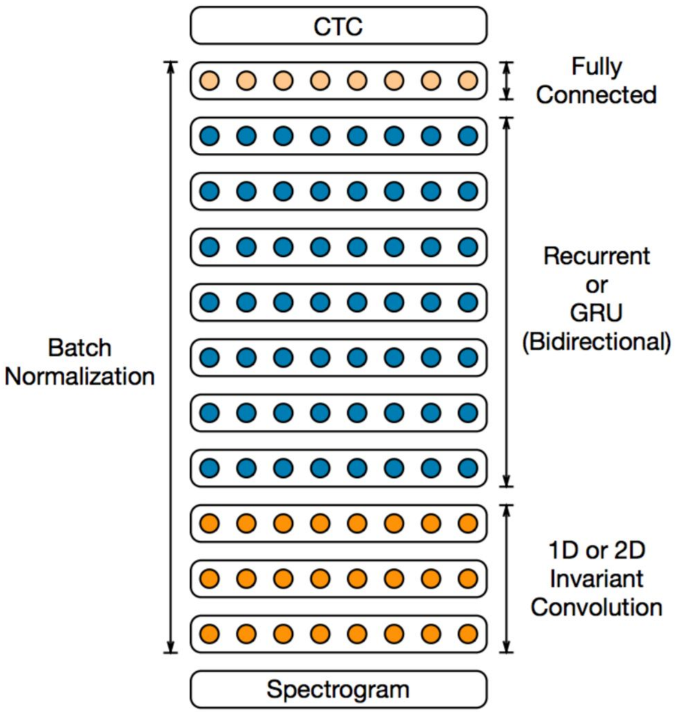


Image: Coates, A. Rao, V. (2016). Speech Recognition and Deep Learning. Retrieved from: https://cs.stanford.edu/~acoates/ba_dls_speech2016.pdf

Connectionist Temporal Classification

Since input is > Output

- Generate at each timestep
- Remove blanks and repeat labels
- Calculate a loss to backprop.

Input Acoustic Features X:	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}
Predicted Alignment:	h	e	l	l	ε	l	o	ε	ε	_	w	o	o	ε	r	l	d
Merge Repeated Predictions:	h	e	l		ε	l	o	ε		_	w	o		ε	r	l	d
Remove *blank* token:	h	e	l			l	o			_	w	o			r	l	d
CTC predicted output Y:	h	e	l	l	o	_	w	o	r	l	d						

Kamath, U., Liu, J., & Whitaker, J. (2019). Deep learning for nlp and speech recognition. Springer International Publishing.

See:

https://pytorch.org/docs/stable/nn.html?highlight=ctc#torch.nn.CTCLoss

Decoding

Generally CTC is bad off the bat (see Deep Speech 2 restults), and much worse than traditional HMM-GMM or HMM-DNN models (e.g. Kaldi TDNN).

However decoding and Language Models help bring it in line.

Approach	WER
Deep Speech 2 (no decoding)	22.83
Deep Speech 2 (4-gram LM, beam size of 512)	5.59
ESPnet (no decoding)	12.34
ESPnet (no LM, beam size of 20)	11.56
Kaldi TDNN (Chap. 8)	4.44

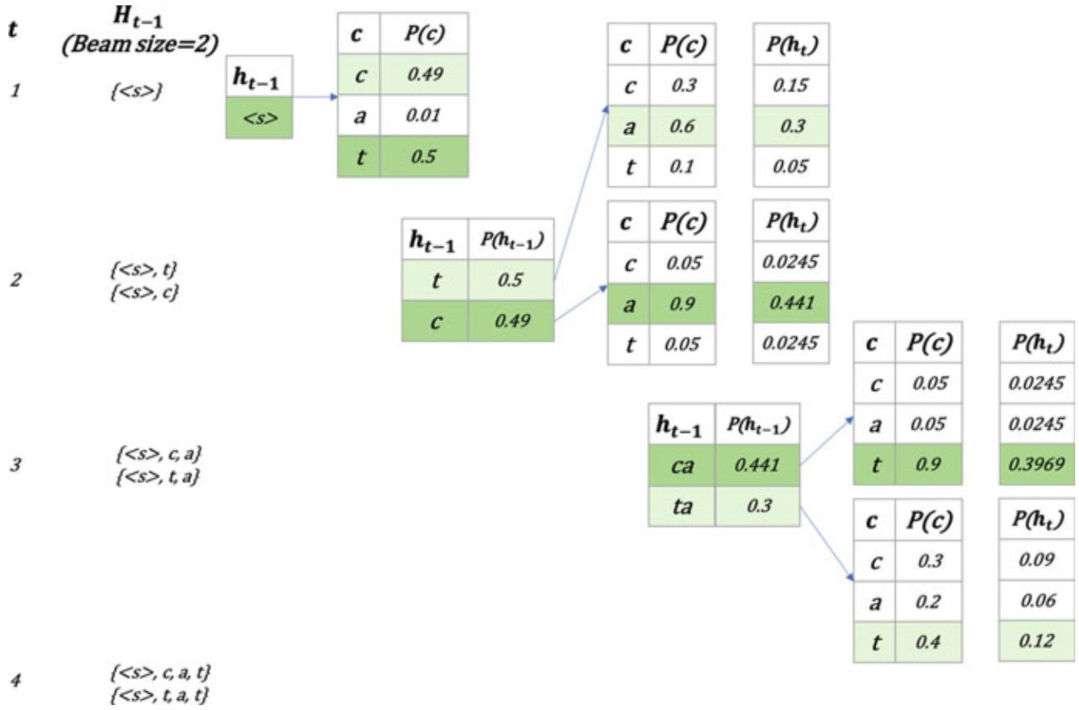
Kamath, U., Liu, J., & Whitaker, J. (2019). Deep learning for nlp and speech recognition. Springer International Publishing.

Best Path

- “Greedy” Decoding
  - Always pick argmax of each time output.
- Can easily miss good results, especially due to the properties of blanks in CTC ex:
  - A_A, AA_ and _AA should all count for same probability, but what if all of these are lower than something else?

Beam Search

Beam search decodes by looking within a top # of paths, potentially allowing you to aggregate paths to find a more optimal solution.



Kamath, U., Liu, J., & Whitaker, J. (2019). Deep learning for nlp and speech recognition. Springer International Publishing.

Improvements to ASR

- Language Models
  - Big improvement by making sure that generated words exist in the language
- Attention
  - Attention Methods can work together with CTC e.g. through Multi-task learning
  - Listen attend and Spell (Chan, Jaitly, Le, and Vinyals, 2016) show that attention methods can emulate the benefit of CTC.
- Embeddings
  - Wav2vec and similar projects aim to emulate the power of word embeddings, but in the context of sound.
- Transformers
  - Newer models attempting to capitalize on better architecture (e.g. Zhou., Dong, Xu, S., & Xu, B. 2018)

References

Chan, W., Jaitly, N., Le, Q., & Vinyals, O. (2016, March). Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4960-4964). IEEE.
Coates, A. Rao, V. (2016). Speech Recognition and Deep Learning. Retrieved from: https://cs.stanford.edu/~acoates/ba_dls_speech2016.pdf
Graves, A., & Jaitly, N. (2014, January). Towards end-to-end speech recognition with recurrent neural networks. In International conference on machine learning (pp. 1764-1772).
Hui, J. (2019, December 26). Speech Recognition Series. Retrieved from https://medium.com/@jonathan_hui/speech-recognition-series-71fd6784551a
Kamath, U., Liu, J., & Whitaker, J. (2019). Deep learning for nlp and speech recognition. Springer International Publishing.
Jaitly, N.. (2017). Natural Language Processing with Deep Learning -Lecture 12: End-to-End Models for Speech Processing Retrieved from https://www.youtube.com/watch?v=3MjIkWxXigM

Schneider, S., Baevski, A., Collobert, R., & Auli, M. (2019). wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*.

Zhou, S., Dong, L., Xu, S., & Xu, B. (2018). Syllable-based sequence-to-sequence speech recognition with the transformer in mandarin chinese. *arXiv preprint arXiv:1804.10752*.