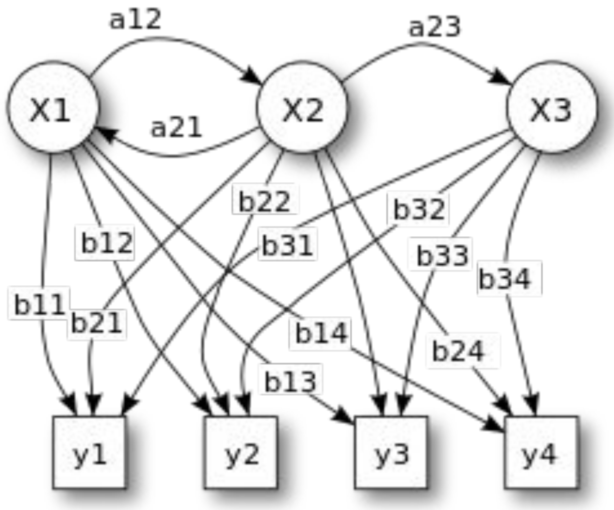
Automatic speech recognition

Overview

1) Models before NN
2) CTC-loss
3) Beam search
4) DeepSpeech
5) Conclusion

Models Before NN

Hidden Markov model (HMM)
Gaussian mixture models (GMM)
Dynamic methods



CTC-loss

$$(\mathbb{R}^m)^T \to (\mathbb{R}^n)^T$$

$$y = a(x)$$

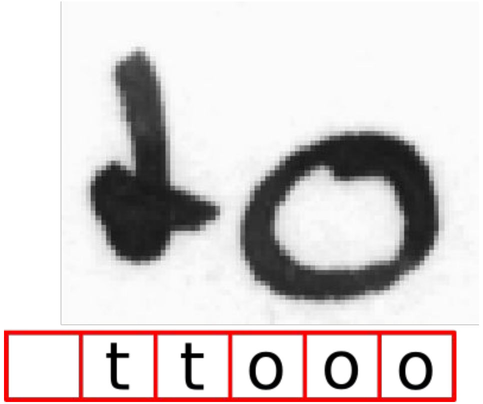
y_k^t - probability of observation unit k at segment t

L - alphabet

$$L' = L \cup \{blank\}$$

$$p(\pi|x) = \prod_{t=1}^{T} y_{\pi_t}^t, \forall \pi \in L'^T$$

CTC-loss



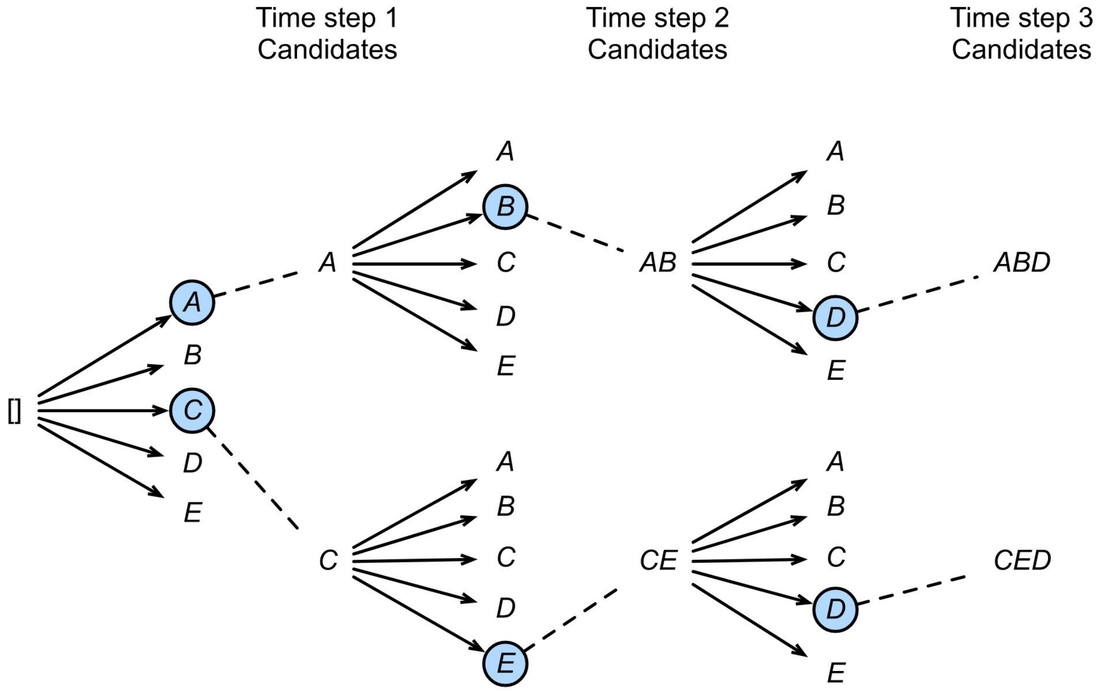
$$p(\mathbf{l}|x) = \sum_{\pi \in \beta^{-1}(\mathbf{l})} p(\pi|x)$$

$$Q_{ML}(S, a) = - \sum_{(x,z) \in S} \log p(z|x)$$

$\beta(\text{“-ttooo”}) \rightarrow \text{“to”}$

$\beta(\text{“tt-oo-ooo”}) \rightarrow \text{“too”}$

Beam Search


Time step 1
Candidates
Time step 2
Candidates
Time step 3
Candidates
[]
A
B
C
D
E
A
A
B
C
D
E
C
A
B
C
D
E
AB
A
B
C
D
E
CE
A
B
C
D
E
ABD
CED

BeamSearch

RNN output	Decoded Transcription
what is the weather like in bostin right now	what is the weather like in boston right now
prime miniter nerenr modi	prime minister narendra modi
arther n tickets for the game	are there any tickets for the game

Table 1: Examples of transcriptions directly from the RNN (left) with errors that are fixed by addition of a language model (right).

$$Q(c) = \log \mathbb{P}(c|x) + \alpha \log \mathbb{P}_{lm}(c) + \beta \text{word\_count}(c)$$

α, β - hyperparameters, $\mathbb{P}_{lm}(c)$ - probability of c according to language model

DeepSpeech

RNN

end-to-end training (spectrogram -> transcript)

robustness to speaker variation and noise

state-of-the-art performance

DeepSpeech

$$\mathcal{X} = \{(x^{(1)}, y^{(1)}), ..., (x^{(n)}, y^{(n)})\}$$

$x^{(i)}$ - time-series with vectors of audio features

$$\hat{y}_t = P(c_t|x),\ c_t \in \{a, b, c, ..., space\}$$

Deep Speech

$$t \in \{1, 2, 3\}$$

$$h_t^{(l)} = g(W^{(l)} h_t^{(l-1)} + b^{(l-1)})$$

$$g(z) = min(max(0, z), 20)$$

Deep Speech

$$h_t^{(f)} = g(W^{(4)} h_t^{(3)} + W_r^{(f)} h_{t-1}^{(f)} + b^{(4)})$$

$$h_t^{(b)} = g(W^{(4)} h_t^{(3)} + W_r^{(b)} h_{t+1}^{(b)} + b^{(4)})$$

$$h_t^{(4)} = h_t^{(f)} + h_t^{(b)}$$

$$h_t^{(5)} = g(W^{(5)} h_t^{(4)} + b^{(5)})$$

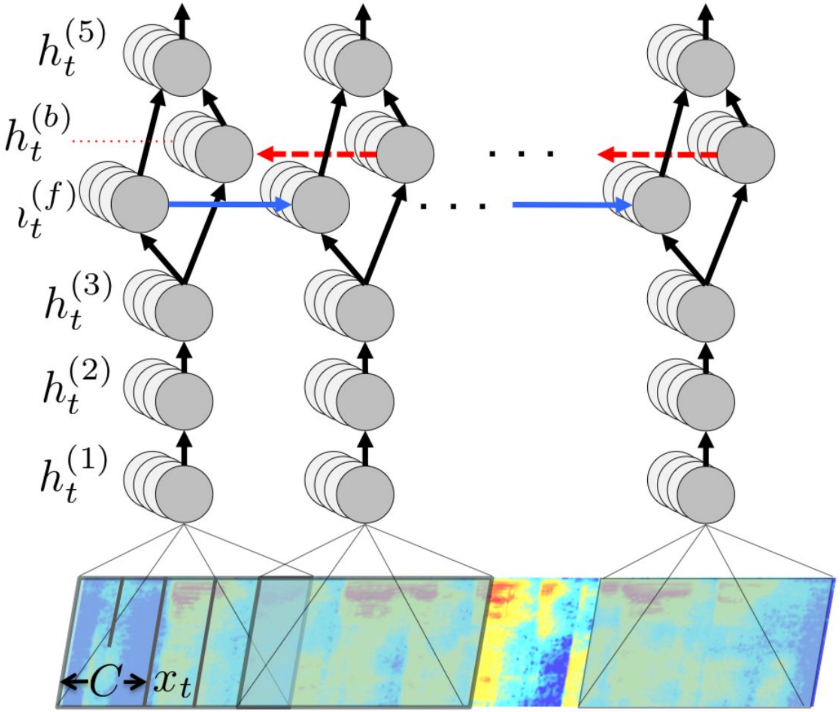


Figure 1: Structure of our RNN model and notation.

$$h_{t,k}^{(6)} = \hat{y}_{t,k} \equiv \mathbb{P}(c_t = k|x) = \frac{\exp(W_k^{(6)} h_t^{(5)} + b_k^{(6)})}{\sum_j \exp(W_j^{(6)} h_t^{(5)} + b_j^{(6)})}$$

DeepSpeech results

Model	SWB	CH	Full
Vesely et al. (GMM-HMM BMMI) [44]	18.6	33.0	25.8
Vesely et al. (DNN-HMM sMBR) [44]	12.6	24.1	18.4
Maas et al. (DNN-HMM SWB) [28]	14.6	26.3	20.5
Maas et al. (DNN-HMM FSH) [28]	16.0	23.7	19.9
Seide et al. (CD-DNN) [39]	16.1	n/a	n/a
Kingsbury et al. (DNN-HMM sMBR HF) [22]	13.3	n/a	n/a
Sainath et al. (CNN-HMM) [36]	11.5	n/a	n/a
Soltau et al. (MLP/CNN+I-Vector) [40]	**10.4**	n/a	n/a
Deep Speech SWB	20.0	31.8	25.9
Deep Speech SWB + FSH	12.6	**19.3**	**16.0**

DeepSpeech results

System	Clean (94)	Noisy (82)	Combined (176)
Apple Dictation	14.24	43.76	26.73
Bing Speech	11.73	36.12	22.05
Google API	6.64	30.47	16.72
wit.ai	7.94	35.06	19.41
Deep Speech	**6.56**	**19.06**	**11.85**

Table 4: Results (%WER) for 5 systems evaluated on the original audio. Scores are reported *only* for utterances with predictions given by all systems. The number in parentheses next to each dataset, e.g. Clean (94), is the number of utterances scored.

Conclusion

CTC-loss - a powerful tool in recognition tasks

Beem search - a way to improve the quality of recognition

DeepSpeech - a breakthrough model in speech recognition task

Sources

1) Deep Speech: Scaling up end-to-end speech recognition

2) Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks