End-to-end Neural Architectures For ASR

Lecture 14

CS 753

Instructor: Preethi Jyothi

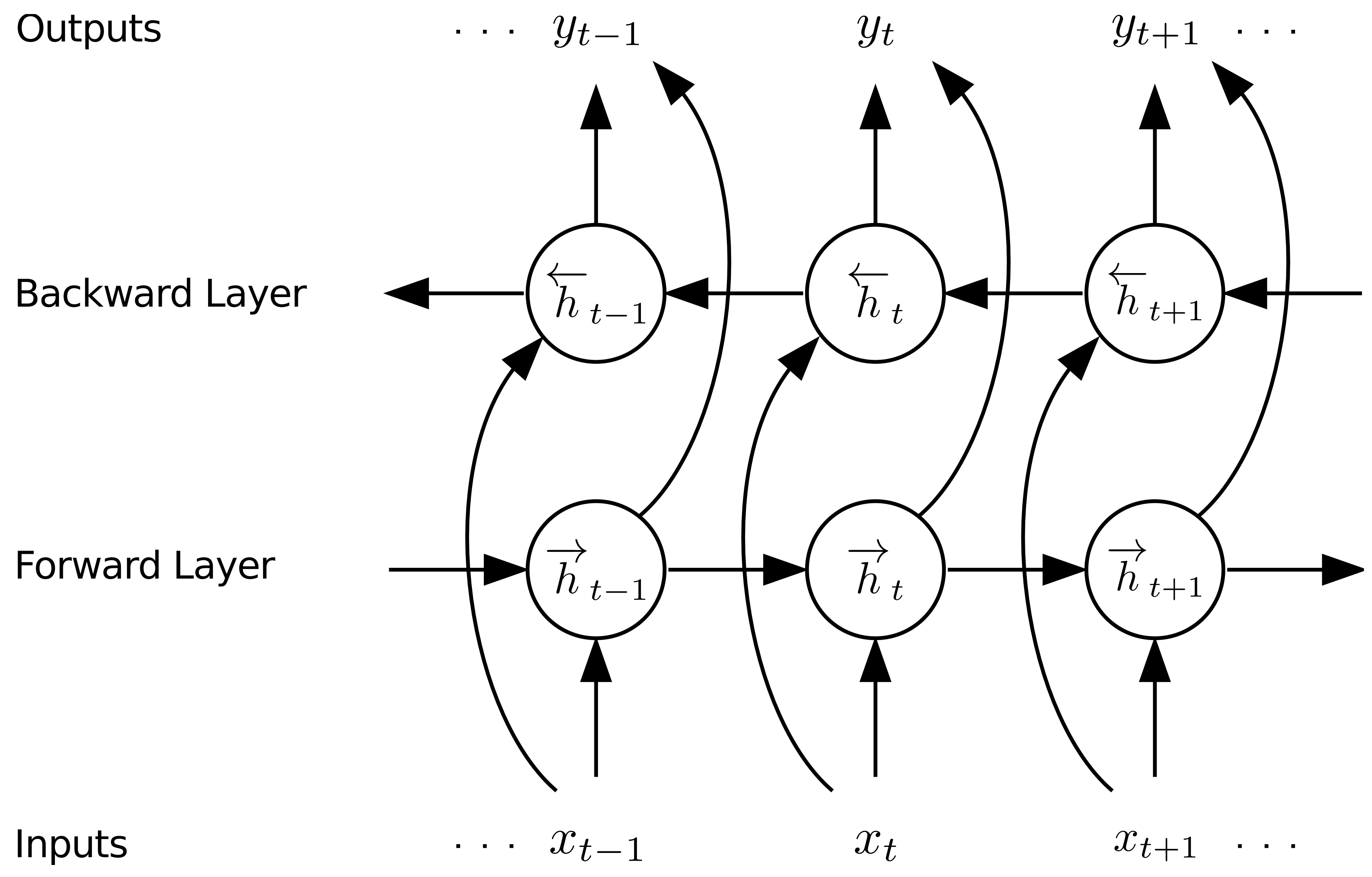
Neural network-based ASR components

- Significant improvements in ASR performance by using neural models for both acoustic models and language models within the ASR pipeline
- However, there are limitations to using neural networks for a single component within such a complex pipeline

Motivation for end-to-end ASR systems

- Limitations:
  - Objective function optimized in neural networks very different from final evaluation metric (i.e. word transcription accuracy)
  - Additionally, frame-level training targets derived from HMM-based alignments
  - Pronunciation dictionaries are used to map from words to phonemes; expensive resource to create
- Can we build a single RNN architecture that represents the entire ASR pipeline?

Network Architecture



$$\overrightarrow{h}_t = \mathcal{H}\left(W_{x\overrightarrow{h}} x_t + W_{\overrightarrow{h}\overrightarrow{h}} \overrightarrow{h}_{t-1} + b_{\overrightarrow{h}}\right)$$

$$\overleftarrow{h}_t = \mathcal{H}\left(W_{x\overleftarrow{h}} x_t + W_{\overleftarrow{h}\overleftarrow{h}} \overleftarrow{h}_{t+1} + b_{\overleftarrow{h}}\right)$$

$$y_t = W_{\overrightarrow{h}y} \overrightarrow{h}_t + W_{\overleftarrow{h}y} \overleftarrow{h}_t + b_o$$

- Input: Acoustic feature vectors. Output: Characters
- Long Short-Term Memory (LSTM) units (with in-built memory cells) are used to implement $\mathcal{H}$ (in eqns above)
- Deep bidirectional LSTMs: Stack multiple bidirectional LSTM layers

Image from: Graves & Jaitley, Towards End-to-End Speech Recognition with Recurrent Neural Networks, ICML 14

Connectionist Temporal Classification (CTC)

- RNNs in ASR, if trained at the frame-level, will typically require alignments between the acoustics and the word sequence during training telling you which label (e.g. phone or character) should be output at each timestep
- A new loss function, Connectionist Temporal Classification (CTC) tries to get around this!
- This is an objective function that allows RNN training without an explicit alignment step: CTC considers all possible alignments

CTC: Prerequisites

- Augment the output vocabulary with an additional “blank” (_) label
- For a given label sequence, there can be multiple alignments: (x, y, z) could correspond to (x, _, y, _, _, z) or (_, x, x, _, y, z)
- Define a 2-step operator B that reduces a label sequence by: first, removing repeating labels and second, removing blanks. B(“x, _, y, _, _, z”) = B(“_, x, x, _, y, z”) = “x, y, z”

CTC Pipeline

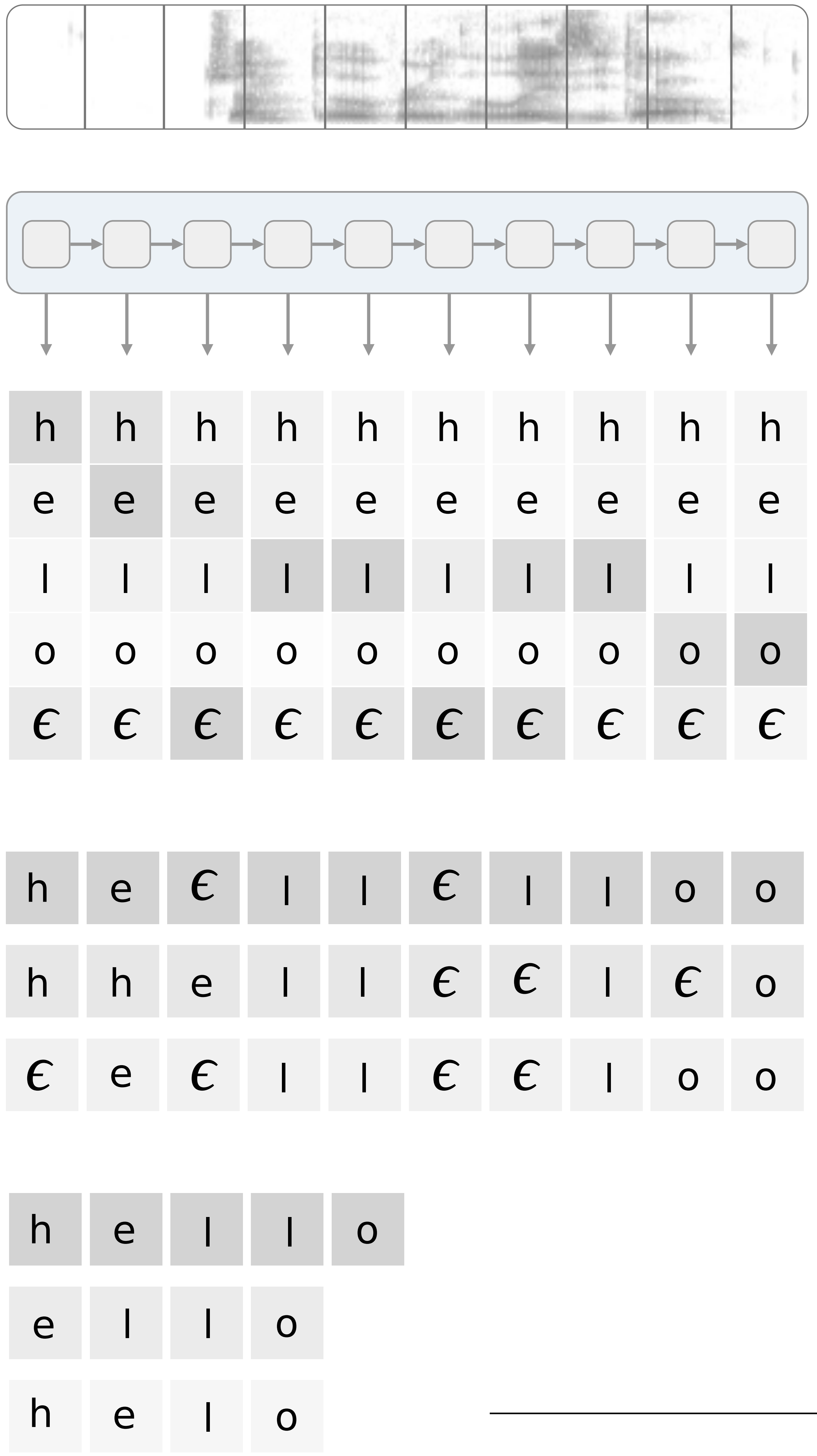


Image from: https://distill.pub/2017/ctc/

CTC Objective Function

- CTC objective function is the probability of an output label sequence y given an utterance x

$$\mathrm{CTC}(x, y) = \Pr(y|x) = \sum_{a \in B^{-1}(y)} \Pr(a|x)$$

- Here, we sum over all possible alignments for y, enumerated by $B^{-1}(y)$

- CTC assumes that $\Pr(a \,|\, x)$ can be computed as $\prod_{t=1}^{T} \Pr(a_t|x)$
  - i.e. CTC assumes that outputs at each time-step are conditionally independent given the input

- Efficient dynamic programming algorithm to compute this loss function and its gradients [GJ14]

[GJ14] Towards End-to-End Speech Recognition with Recurrent Neural Networks, ICML 14

Illustration: Dynamic Programming Algorithm

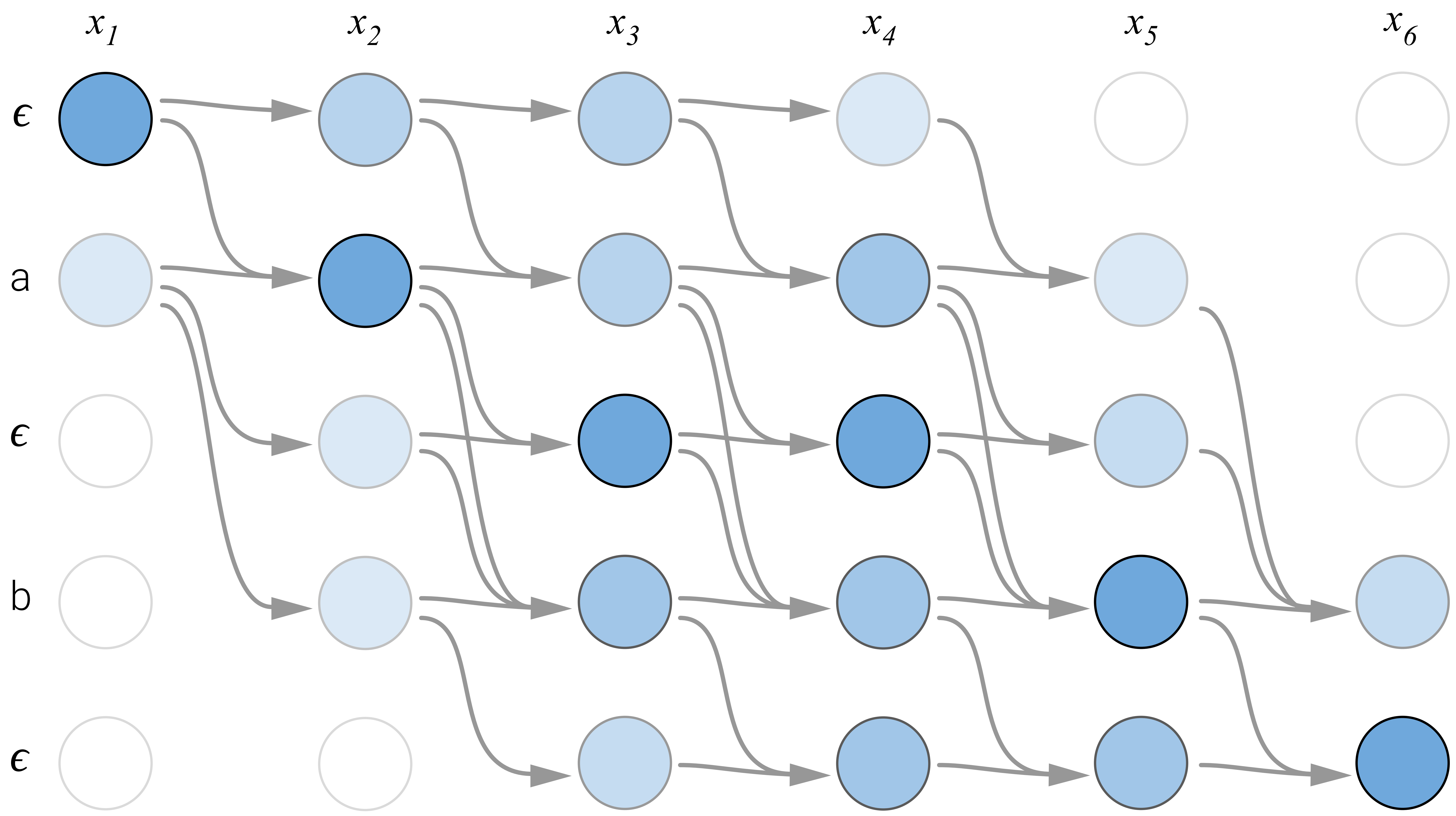


Image from: https://distill.pub/2017/ctc/

Decoding

- Pick the single most probable output at every time step

$$\arg\max_{y} \Pr(y|x) \approx B(\arg\max_{a} Pr(a|x))$$

- Use a beam search algorithm to integrate a dictionary and a language model

- Beam search will be covered in more detail next week

Experimental Results

Table 1. **Wall Street Journal Results.** All scores are word error rate/character error rate (where known) on the evaluation set. ‘LM’ is the Language model used for decoding. ‘14 Hr’ and ‘81 Hr’ refer to the amount of data used for training.

SYSTEM	**LM**	**14 HR**	**81 HR**
RNN-CTC	NONE	74.2/30.9	30.1/9.2
RNN-CTC	DICTIONARY	69.2/30.0	24.0/8.0
RNN-CTC	MONOGRAM	25.8	15.8
RNN-CTC	BIGRAM	15.5	10.4
RNN-CTC	TRIGRAM	13.5	8.7
BASELINE	NONE	—	—
BASELINE	DICTIONARY	56.1	51.1
BASELINE	MONOGRAM	23.4	19.9
BASELINE	BIGRAM	11.6	9.4
BASELINE	TRIGRAM	9.4	7.8
COMBINATION	TRIGRAM	—	6.7

Table from: Graves & Jaitley, Towards End-to-End Speech Recognition with Recurrent Neural Networks, ICML 14

Sample char-level transcripts

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*

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*

target: *ALL THE EQUITY RAISING IN MILAN GAVE THAT STOCK MARKET INDIGESTION LAST YEAR*
output: *ALL THE EQUITY RAISING IN MULONG GAVE THAT STACRK MARKET IN TO JUSTIAN LAST YEAR*

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*

Network Outputs

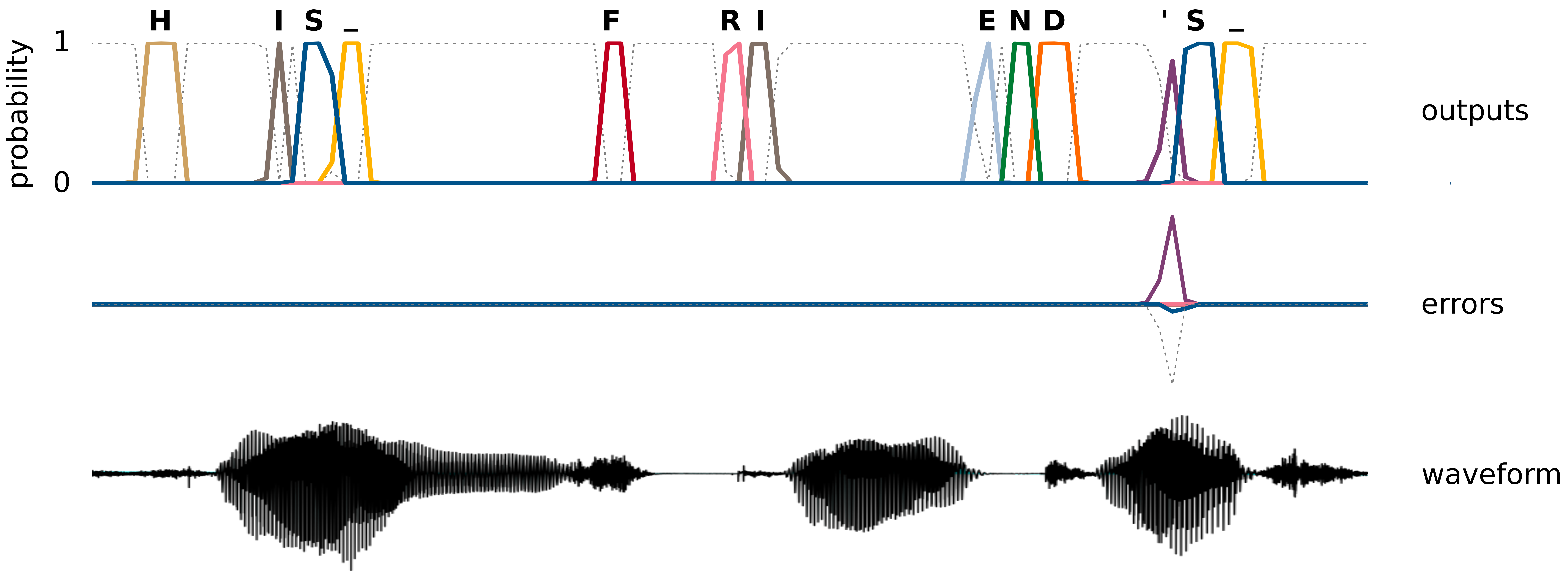


Image from: Graves & Jaitley, Towards End-to-End Speech Recognition with Recurrent Neural Networks, ICML 14

Another end-to-end system

- Decoding is still at the word level. Out-of-vocabulary (OOV) words cannot be handled.
- Build a system that is trained and decoded entirely at the character-level [M et al.].
  - This would enable the transcription of OOV words, disfluencies, etc.
- Shows results on the Switchboard task. Matches a GMM-HMM baseline system but underperforms compared to an HMM-DNN baseline.

[M et al.]:Maas et al., “Lexicon Free Conversational Speech Recognition with Neural Networks”, NAACL 15

Model Specifics

- Approach consists of two neural models:
  - A deep bidirectional RNN (DBRNN) mapping acoustic features to character sequences (Trained using CTC.)
  - A neural network character language model

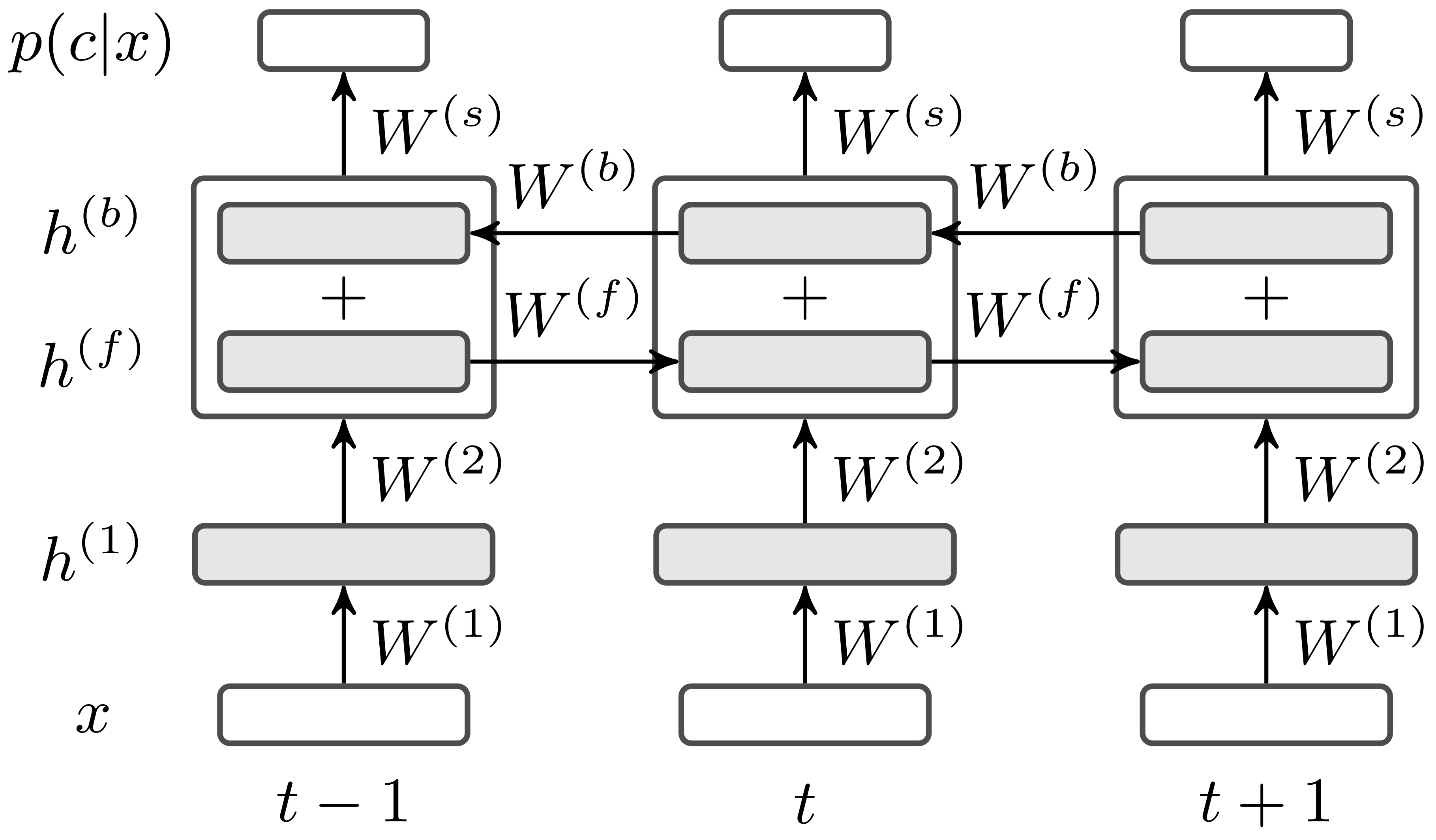


Image from Maas et al., “Lexicon Free Conversational Speech Recognition with Neural Networks”, NAACL 15

Decoding

- Simplest form: Decode without any language model
- Beam Search decoding:
  - Combine DBRNN outputs with a char-level language model
  - Char-level language model applied at every time step (unlike word models)
  - Circumvents the issue of handling OOV words during decoding
  - More about beam search in the coming week.

Experimental Results

Method	CER	EV	CH	SWBD
HMM-GMM	23.0	29.0	36.1	21.7
HMM-DNN	17.6	21.2	27.1	15.1
HMM-SHF	NR	NR	NR	12.4
CTC no LM	27.7	47.1	56.1	38.0
CTC+5-gram	25.7	39.0	47.0	30.8
CTC+7-gram	24.7	35.9	43.8	27.8
CTC+NN-1	24.5	32.3	41.1	23.4
CTC+NN-3	24.0	30.9	39.9	21.8
CTC+RNN	24.9	33.0	41.7	24.2
CTC+RNN-3	24.7	30.8	40.2	21.4

Table 1: Character error rate (CER) and word error rate results on the Eval2000 test set. We report word error rates on the full test set (EV) which consists of the Switchboard (SWBD) and CallHome (CH) subsets. As baseline systems we use an HMM-GMM system and HMM-DNN system. We evaluate our DBRNN trained using CTC by decoding with several character-level language models: 5-gram, 7-gram, densely connected neural networks with 1 and 3 hidden layers (NN-1, and NN-3), as well as recurrent neural networks s with 1 and 3 hidden layers. We additionally include results from a state-of-the-art HMM-based system (HMM-DNN-SHF) which does not report performance on all metrics we evaluate (NR).

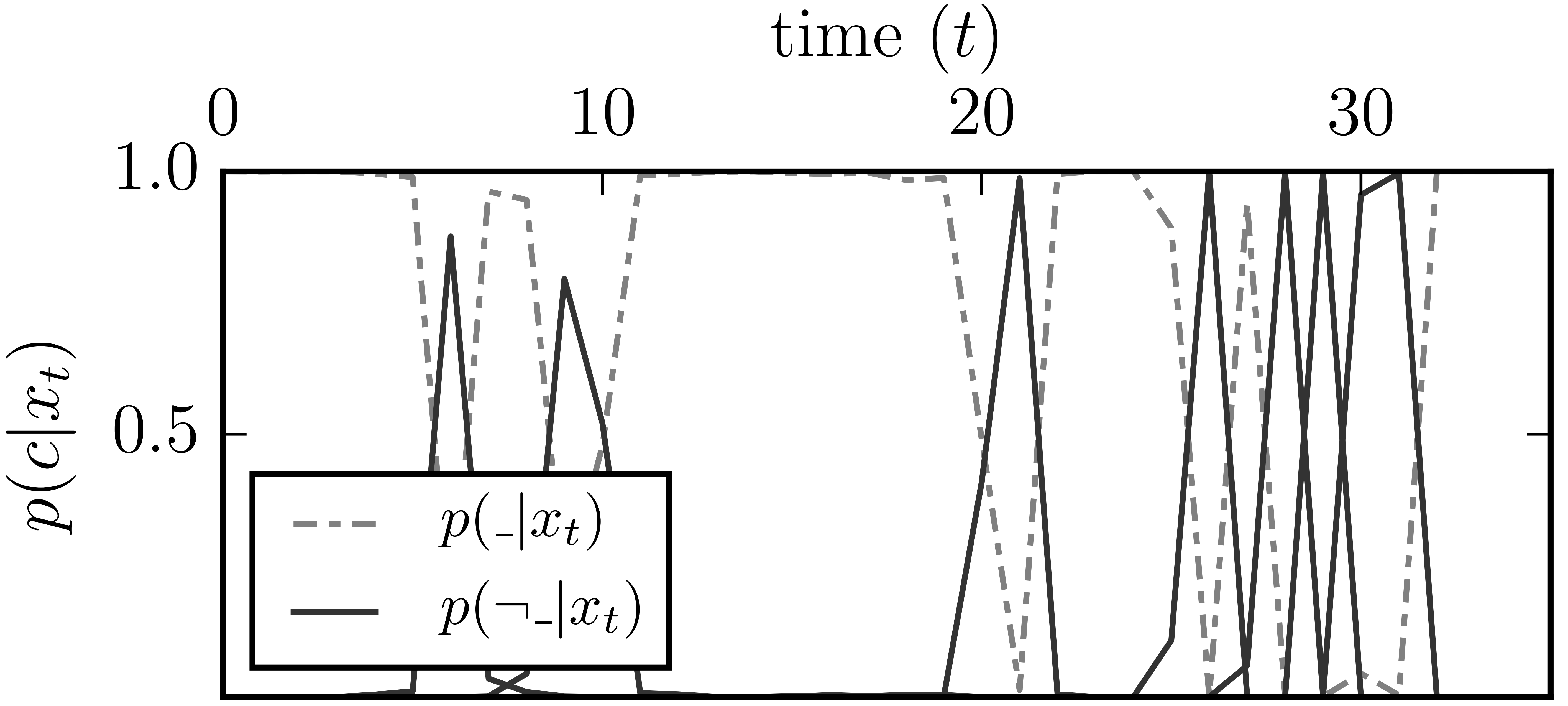
Image from Maas et al., “Lexicon Free Conversational Speech Recognition with Neural Networks”, NAACL 15

Sample Test Utterances

#	Method	Transcription
(1)	Truth	yeah i went into the i do not know what you think of *fidelity* but
	HMM-GMM	yeah when the i don't know what you think of fidel it even them
	CTC+CLM	yeah i went to i don't know what you think of fidelity but um
(2)	Truth	no no speaking of weather do you carry a altimeter slash *barometer*
	HMM-GMM	no i'm not all being the weather do you uh carry a uh helped emitters last brahms her
	CTC+CLM	no no beating of whether do you uh carry a uh a time or less barometer
(3)	Truth	i would ima- well yeah it is i know you are able to stay home with them
	HMM-GMM	i would amount well yeah it is i know um you're able to stay home with them
	CTC+CLM	i would ima- well yeah it is i know uh you're able to stay home with them

Image from Maas et al., "Lexicon Free Conversational Speech Recognition with Neural Networks", NAACL 15

Analysing character probabilities



s: `_____o__hh__________ ____y_eahh___`

$\kappa(s)$: `oh yeah`

Image from Maas et al., “Lexicon Free Conversational Speech Recognition with Neural Networks”, NAACL 15