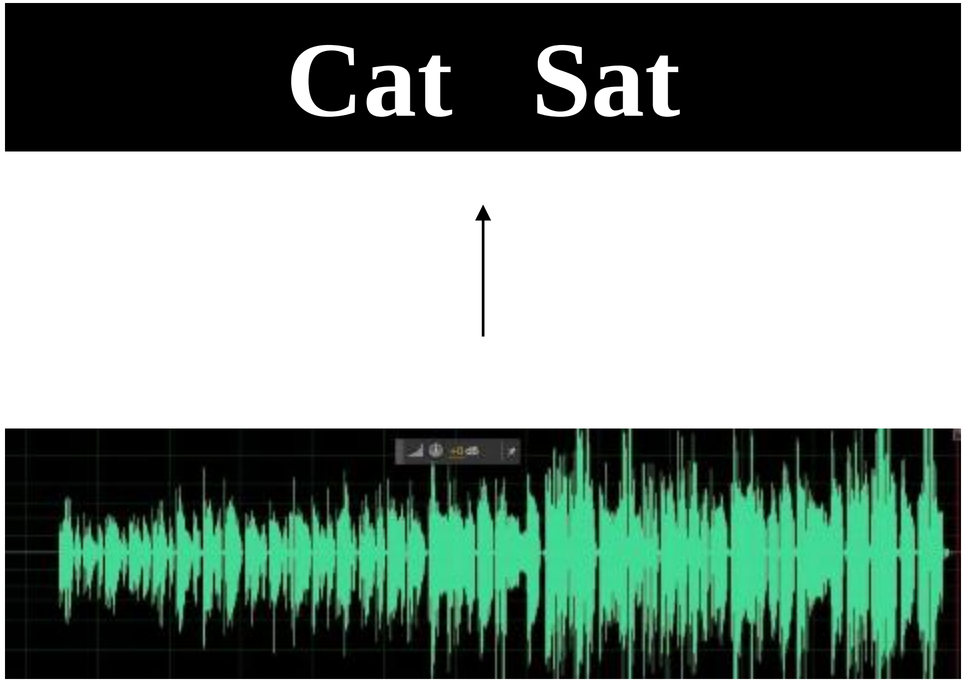
Connectionist **T**emporal **C**lassification:

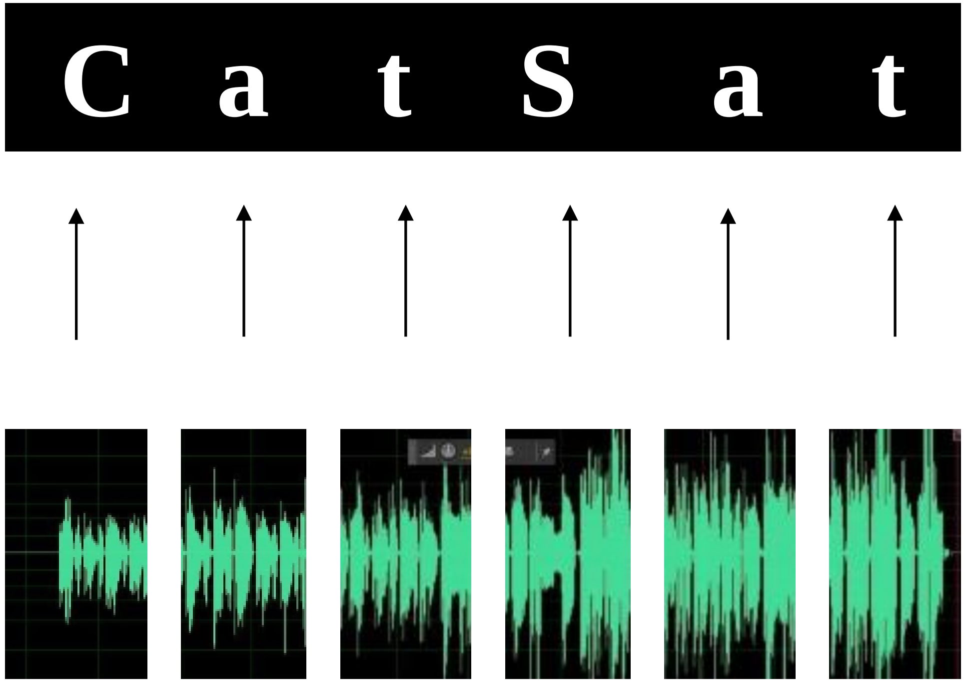
Labelling Unsegmented Sequence Data with Recurrent Neural Networks

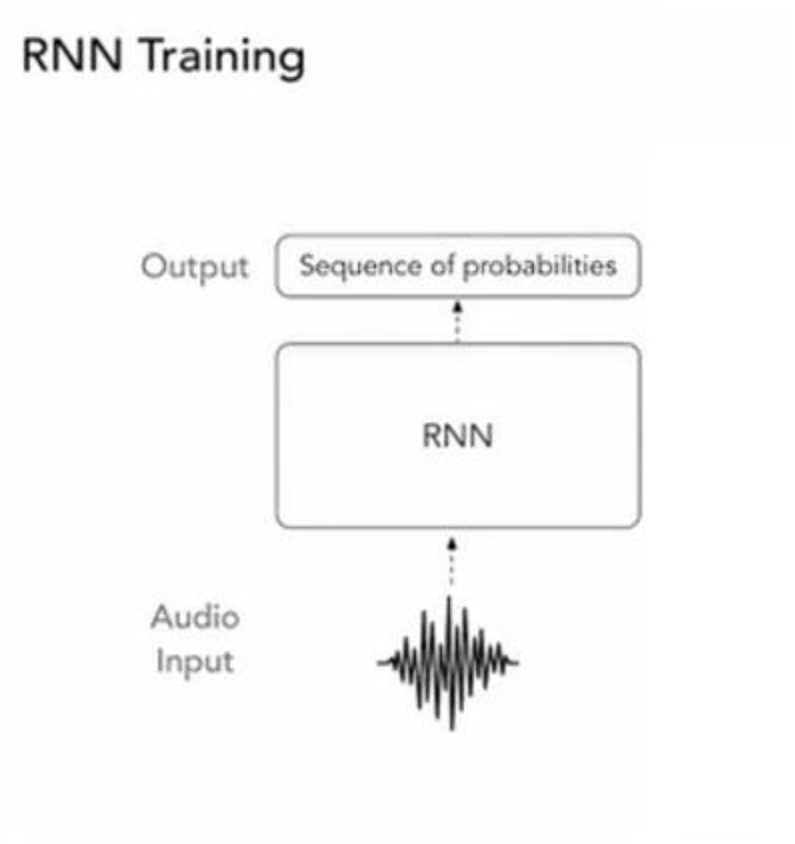
0 Background



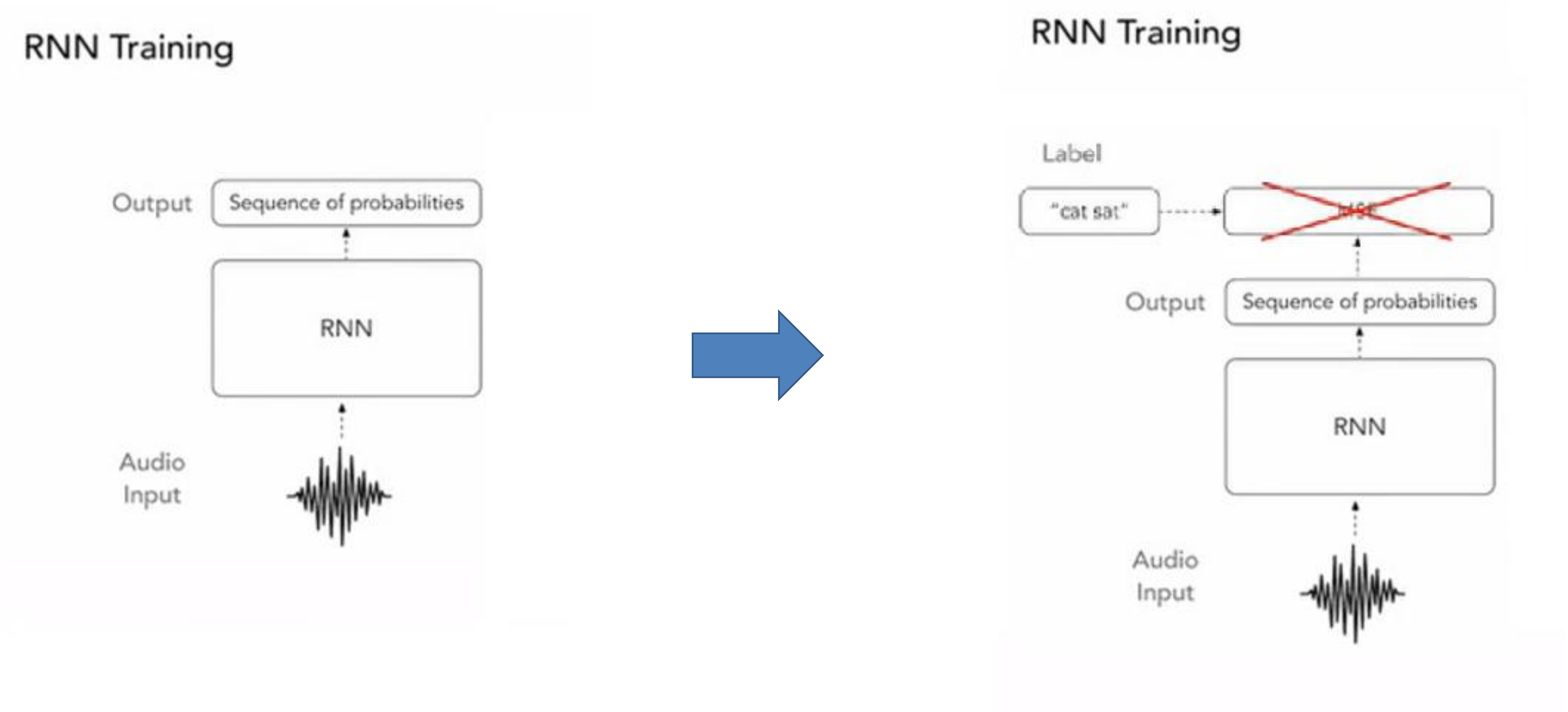
Many real-world sequence learning tasks require the prediction of sequences of labels from noisy, unsegmented input data. In speech recognition, for example, an acoustic signal is transcribed into words or sub-word units. Recurrent neural networks (RNNs) are powerful sequence learners that would seem well suited to such tasks. However, because they require pre-segmented training data, and post-processing to transform their outputs into label sequences, their applicability has so far been limited. This paper presents a novel method for training RNNs to label unsegmented sequences directly, thereby solving both problems. An experiment on the TIMIT speech corpus demonstrates its advantages over both a baseline HMM and a hybrid HMM-RNN.

0 Background

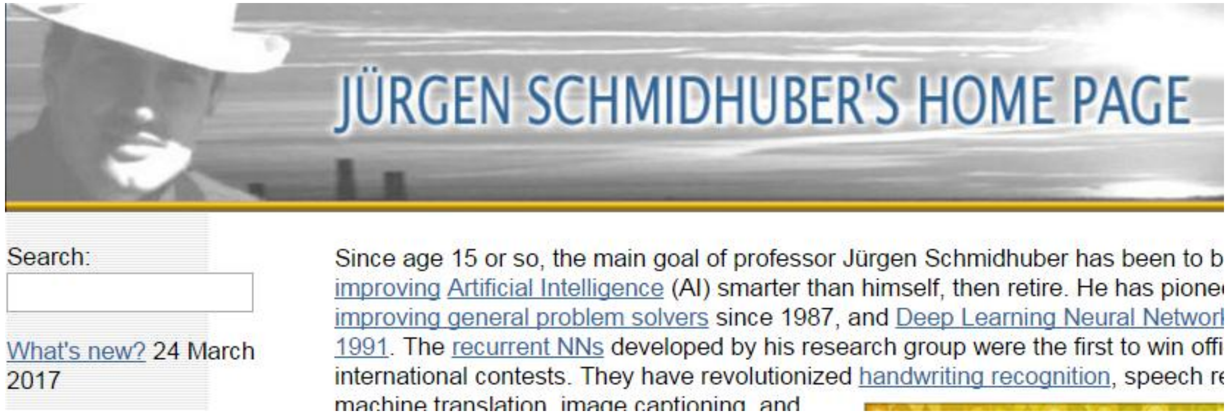




0 Background



1 Introduction

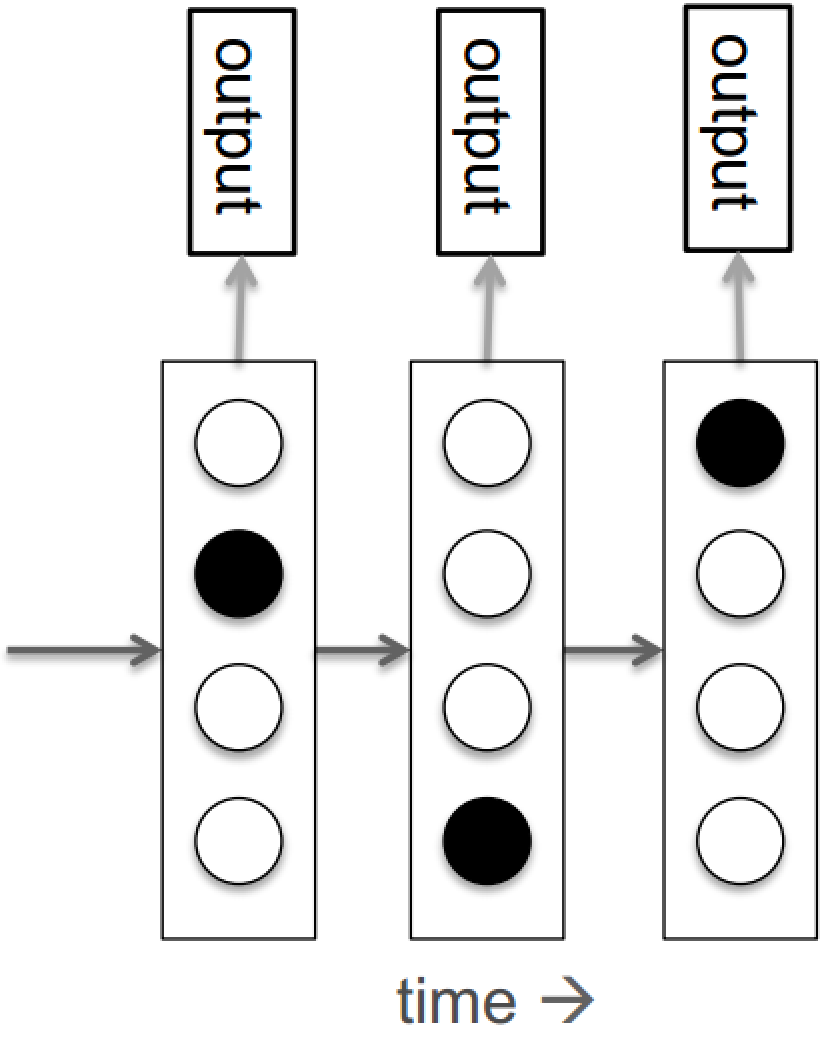


Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks

1 Introduction

（1）混合声学模型

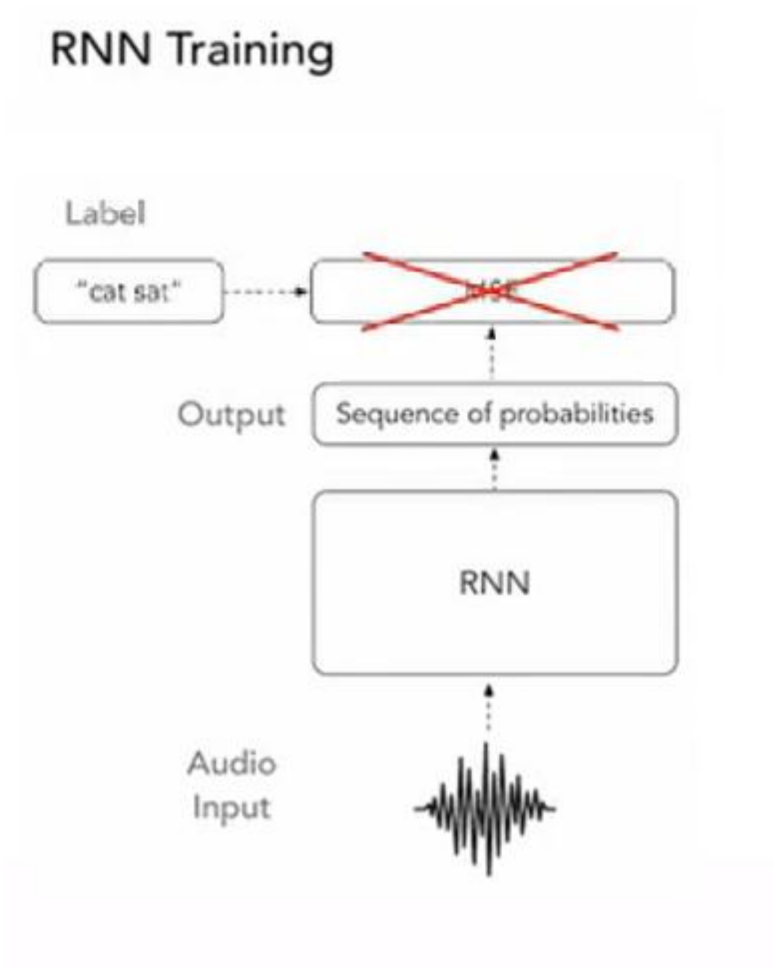
GMM-HMM 混合高斯-隐马尔科夫模型
GMM-HMM 深度神经网络-隐马尔科夫模型
RNN-HMM (Hybrid方法) 深度循环神经网络-隐马尔科夫模型

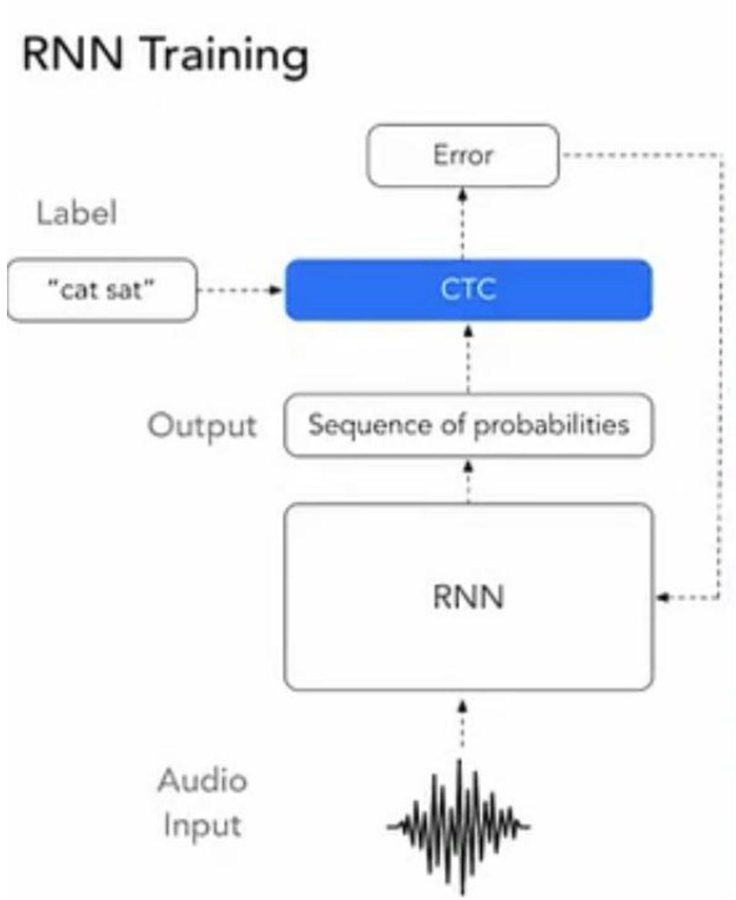


（2）端到端的声学模型

LSTM-CTC 连接时序分类-长短时记忆模型

1 Introduction

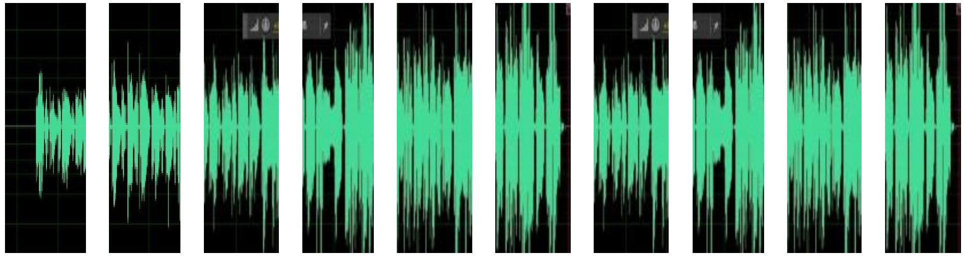




2 Temporal Classification

原始样本

隐式分割：Temporal
无明确意义(比如时间)的采样



C C A A T S A A T T

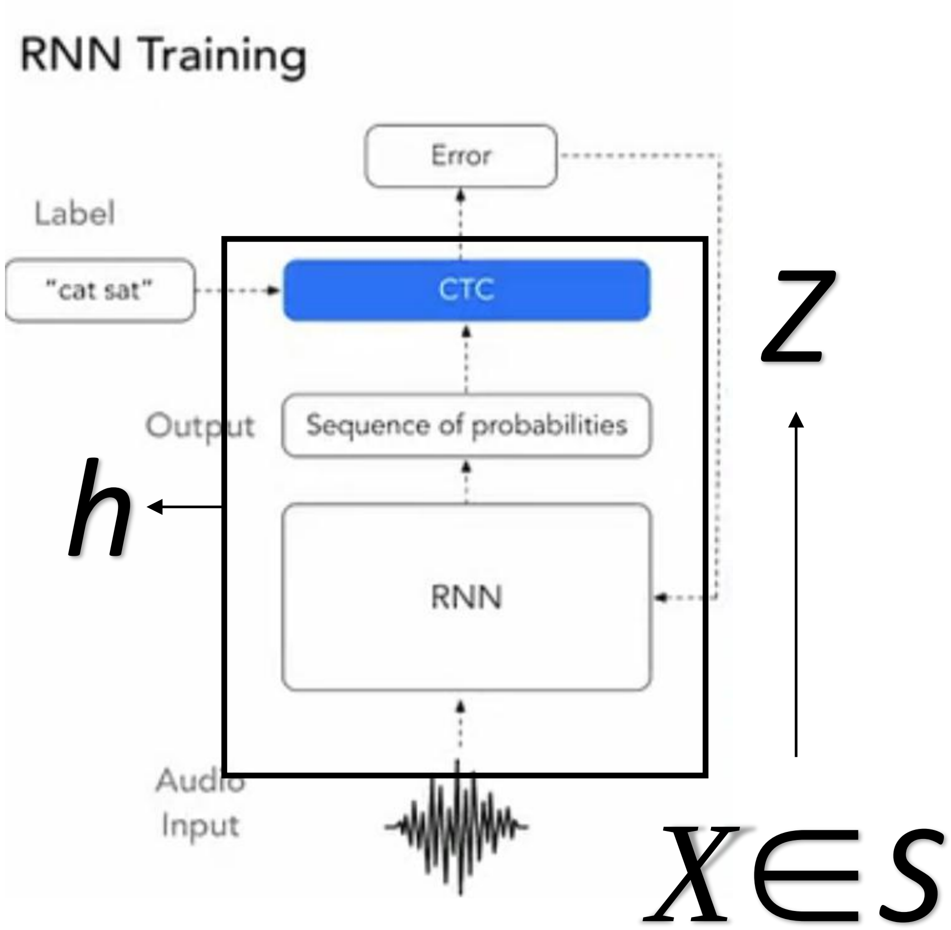
Temporal Classification

显示分割：FrameWise
对应label的准确分割

C A T S A T

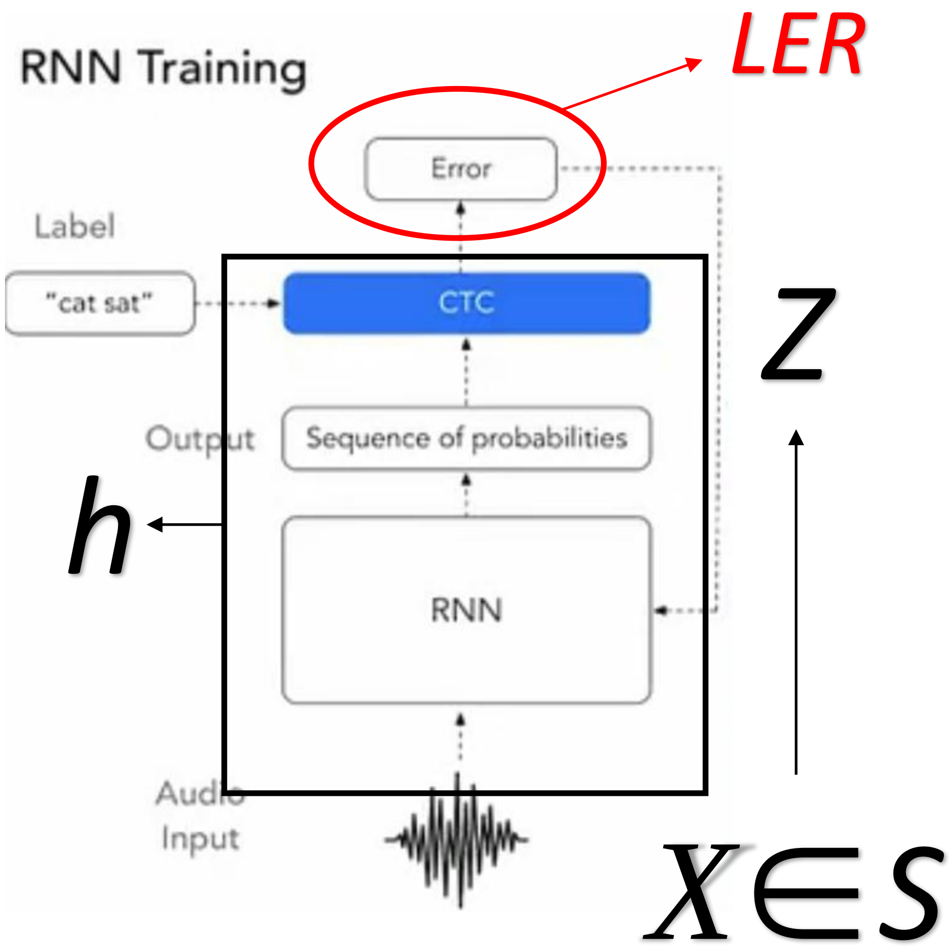
FrameWise Classification

2 Temporal Classification



The aim is to use S to train a temporal classifier $h : \mathcal{X} \mapsto \mathcal{Z}$ to classify previously unseen input sequences in a way that minimises some task specific error measure.

2 Temporal Classification



In this paper, we are interested in the following error measure: given a test set $S' \subset \mathcal{D}_{\mathcal{X}\times\mathcal{Z}}$ disjoint from S, define the *label error rate* (LER) of a temporal classifier h as the mean normalised edit distance between its classifications and the targets on S', i.e.

$$\boxed{LER(h, S') = \frac{1}{|S'|} \sum_{(\mathbf{x},\mathbf{z})\in S'} \frac{ED(h(\mathbf{x}), \mathbf{z})}{|\mathbf{z}|} \qquad (1)}$$

This is a natural measure for tasks (such as speech or handwriting recognition) where the aim is to minimise the rate of transcription mistakes.

编辑距离，简单讲就是：

2 Temporal Classification

Edit Distance:编辑距离

where $ED(\mathbf{p}, \mathbf{q})$ is the edit distance between two sequences $\mathbf{p}$ and $\mathbf{q}$ — i.e. the minimum number of insertions, substitutions and deletions required to change $\mathbf{p}$ into $\mathbf{q}$.

例如将kitten转成sitting：

kitten->sitten （k→s）

sitten->sittin （e→i）

sittin->sitting （插入g）

俄罗斯科学家Vladimir Levenshtein在1965年提出这个概念。

3 Connectionist Temporal Classification

3.1. From Network Outputs to Labellings

$$(\mathbb{R}^m)^T \mapsto (\mathbb{R}^n)^T. \text{ Let } \mathbf{y} = \mathcal{N}_w(\mathbf{x})$$

$$p(\pi|\mathbf{x}) = \prod_{t=1}^{T} y^t_{\pi_t}, \ \forall \pi \in L'^T. \quad (2)$$

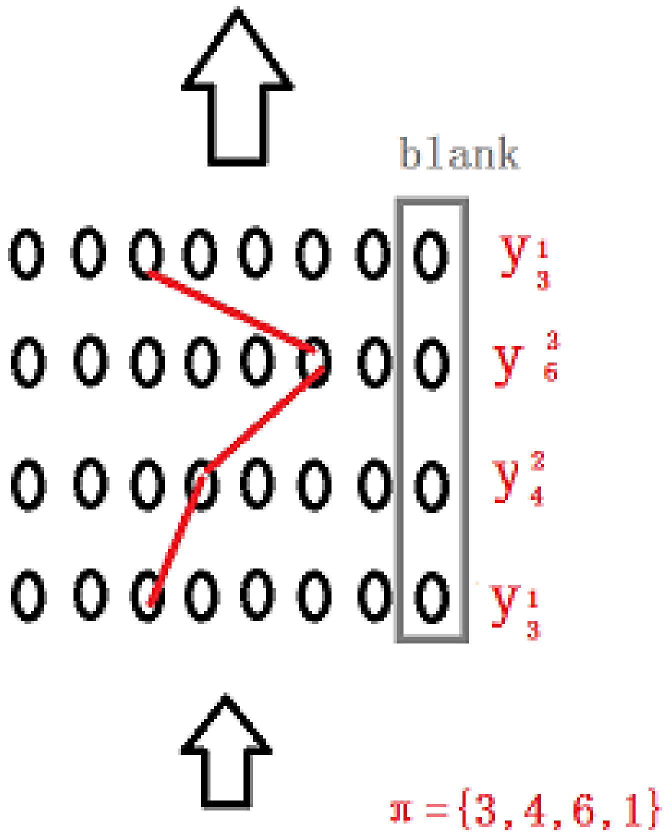
More formally, for an input sequence $\mathbf{x}$ of length T, define a recurrent neural network with m inputs, n outputs and weight vector w as a continuous map $\mathcal{N}_w : (\mathbb{R}^m)^T \mapsto (\mathbb{R}^n)^T$. Let $\mathbf{y} = \mathcal{N}_w(\mathbf{x})$ be the sequence of network outputs, and denote by y^t_k the activation of output unit k at time t. Then y^t_k is interpreted as the probability of observing label k at time t, which defines a distribution over the set L'^T of length T sequences over the alphabet $L' = L \cup \{blank\}$:

3 Connectionist Temporal Classification

1 to 1 （one timestep to one label）

$$(\mathbb{R}^m)^T \mapsto (\mathbb{R}^n)^T. \text{ Let } \mathbf{y} = \mathcal{N}_w(\mathbf{x})$$

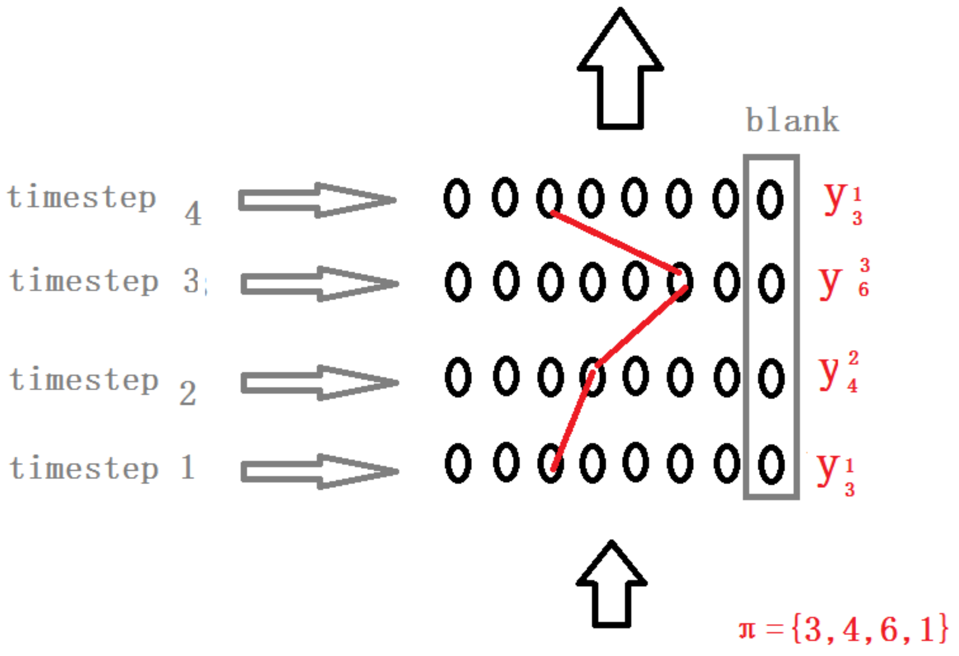
$$p(\pi|\mathbf{x}) = \prod_{t=1}^{T} y^t_{\pi_t}, \ \forall \pi \in L'^T. \qquad (2)$$



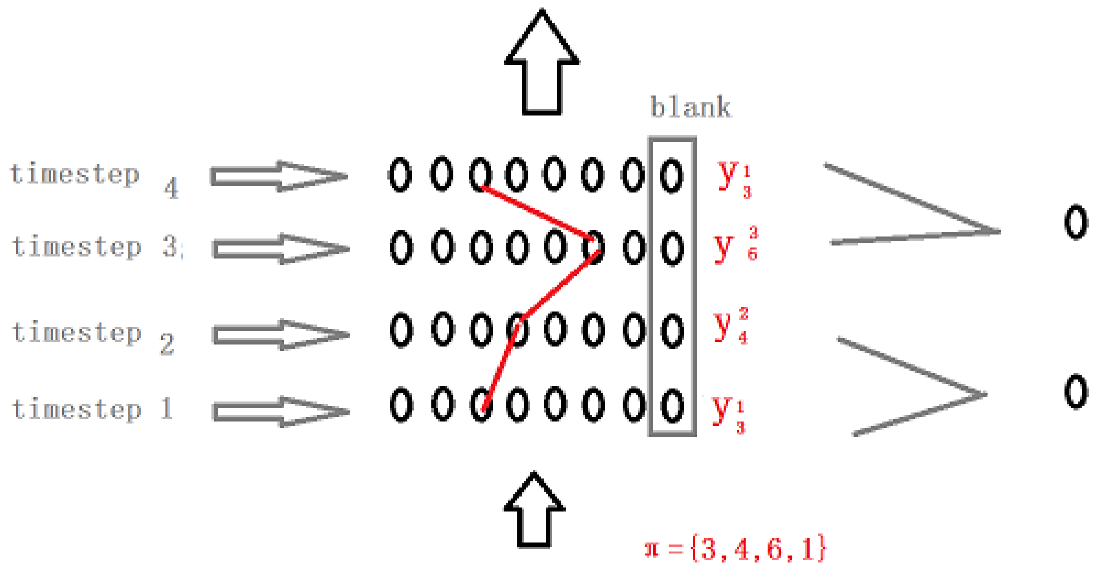
1 to 1 （one timestep to one label）

3 Connectionist Temporal Classification

1 to 1
（one timestep to one label）



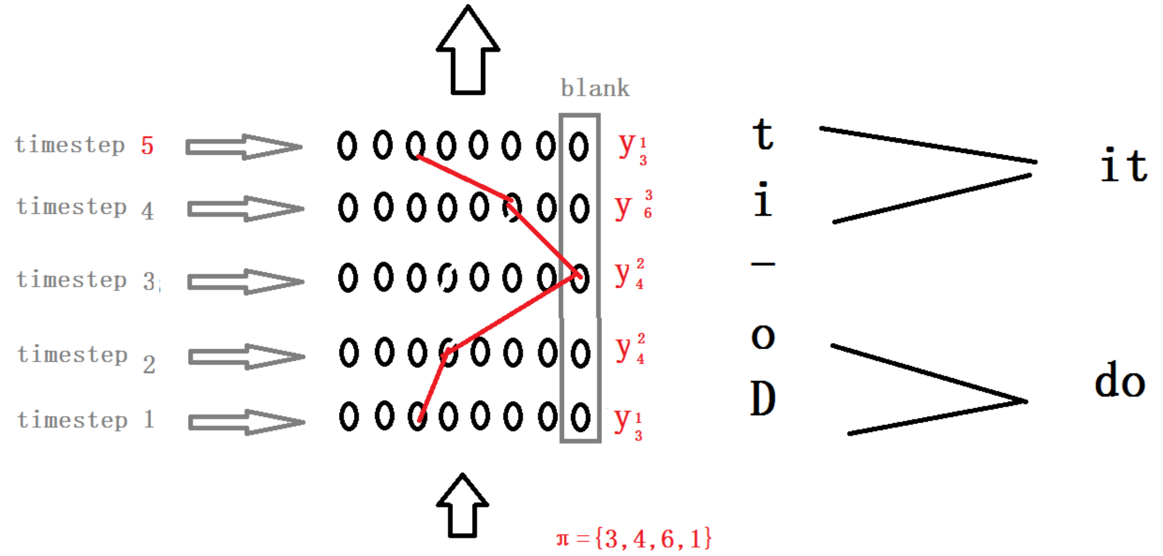
many to 1
（one timestep to one label）



3 Connectionist Temporal Classification

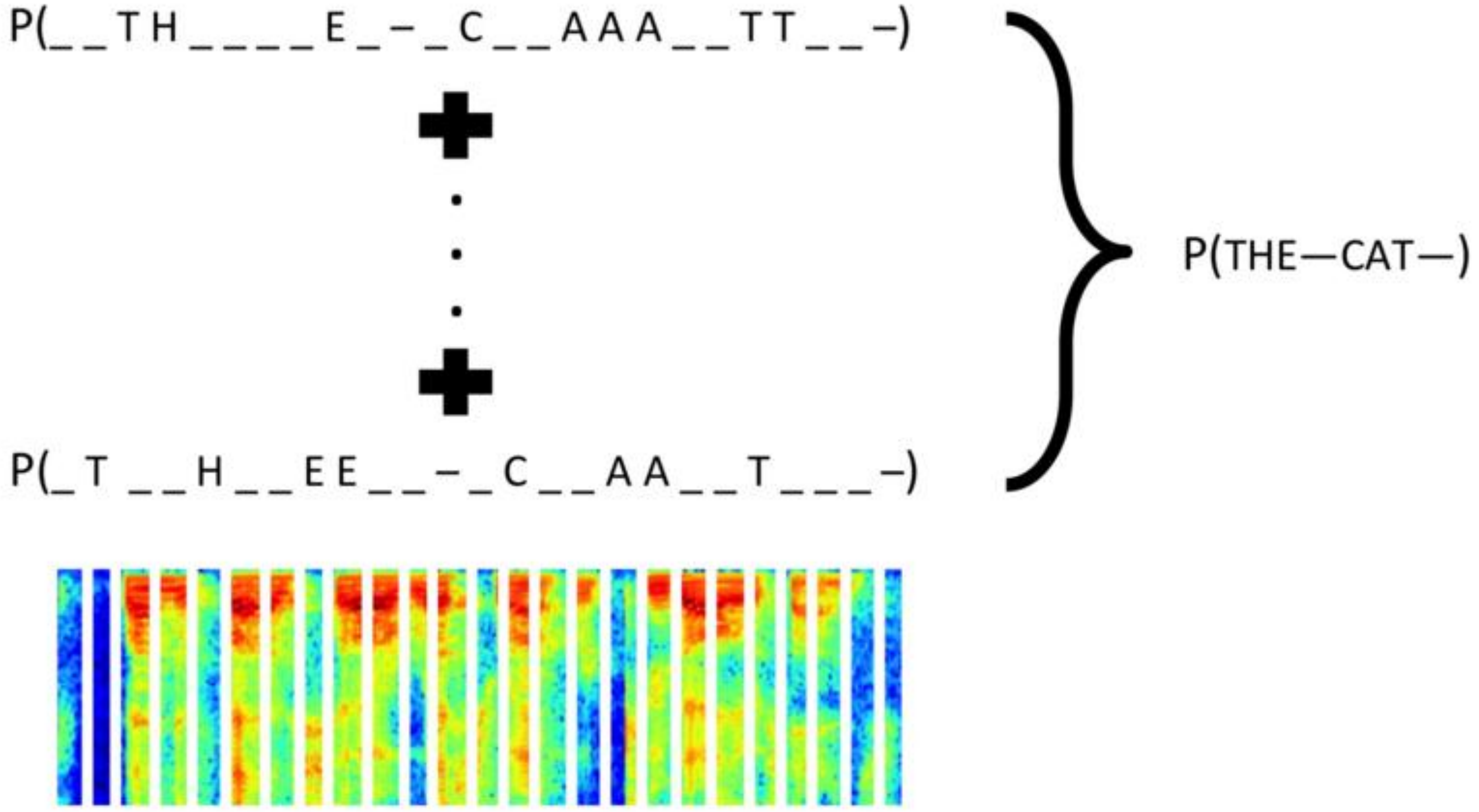
many to 1

（one timestep to one label）



$$\beta(\text{-aa—abb}) = \beta(\text{a-ab-}) = \text{aab}$$

3 Connectionist Temporal Classification



3 Connectionist Temporal Classification

many to 1

（one timestep to one label）

$$p(\mathbf{l}|\mathbf{x}) = \sum_{\pi \in \mathcal{B}^{-1}(\mathbf{l})} p(\pi|\mathbf{x}). \qquad (3)$$

$$h(\mathbf{x}) \approx \mathcal{B}(\pi^*)$$
$$\text{where } \pi^* = \arg\max_{\pi \in N^t} p(\pi|\mathbf{x}).$$

3 Connectionist Temporal Classification

3.2. Constructing the Classifier

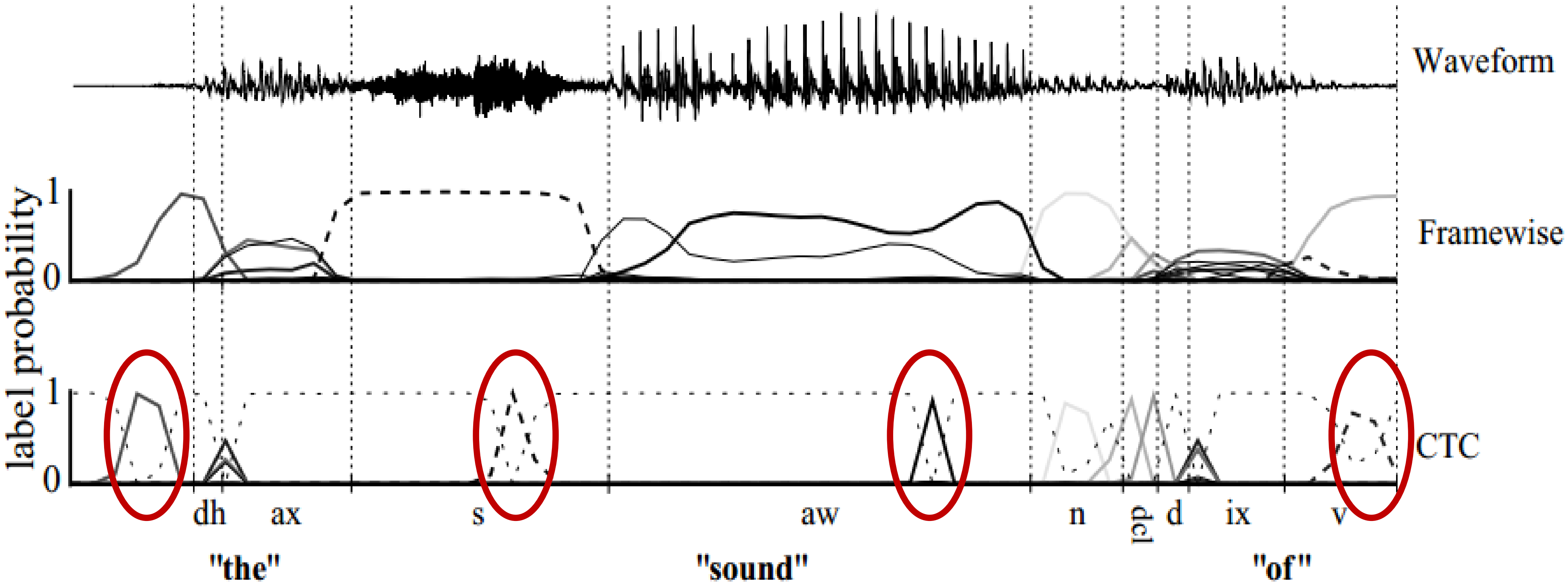
Given the above formulation, the output of the classifier should be the most probable labelling for the input sequence:

$$h(\mathbf{x}) = \arg\max_{\mathbf{l} \in L^{\leq T}} p(\mathbf{l}|\mathbf{x}).$$

Using the terminology of HMMs, we refer to the task of finding this labelling as *decoding*. Unfortunately, we do not know of a general, tractable decoding algorithm for our system. However the following two approximate methods give good results in practice.

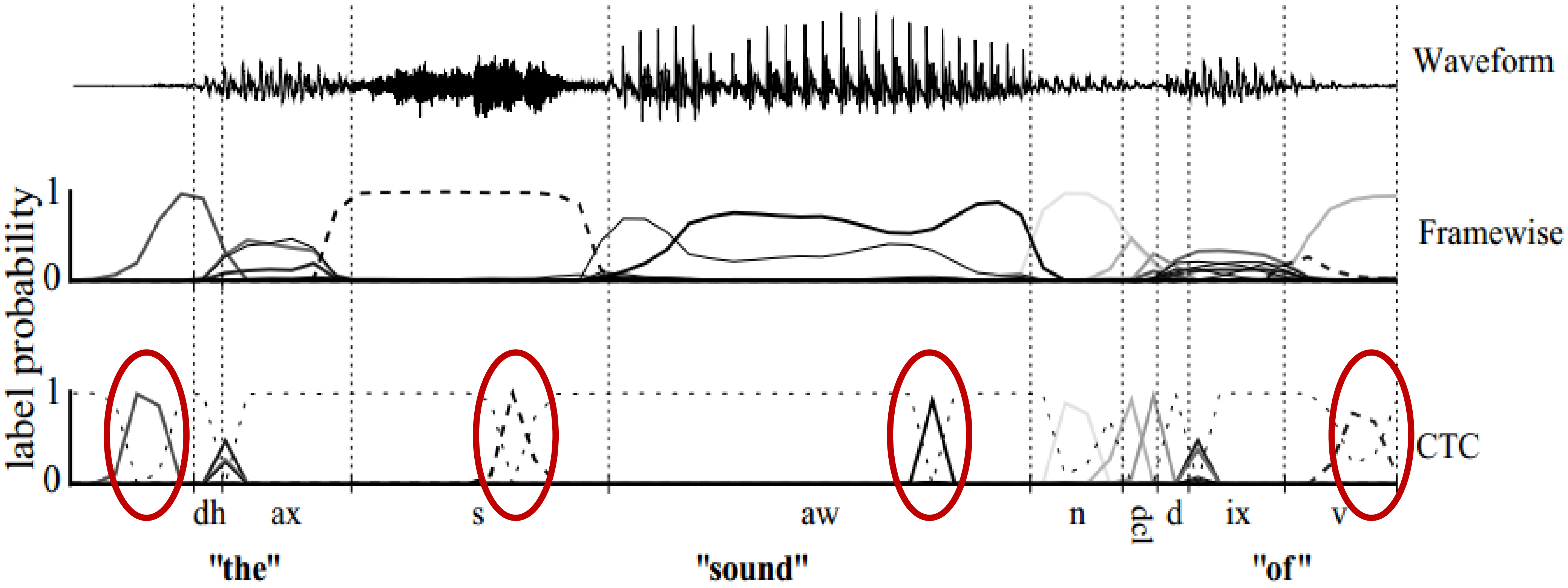
3 Connectionist Temporal Classification

3.2 **Prefix search decoding**

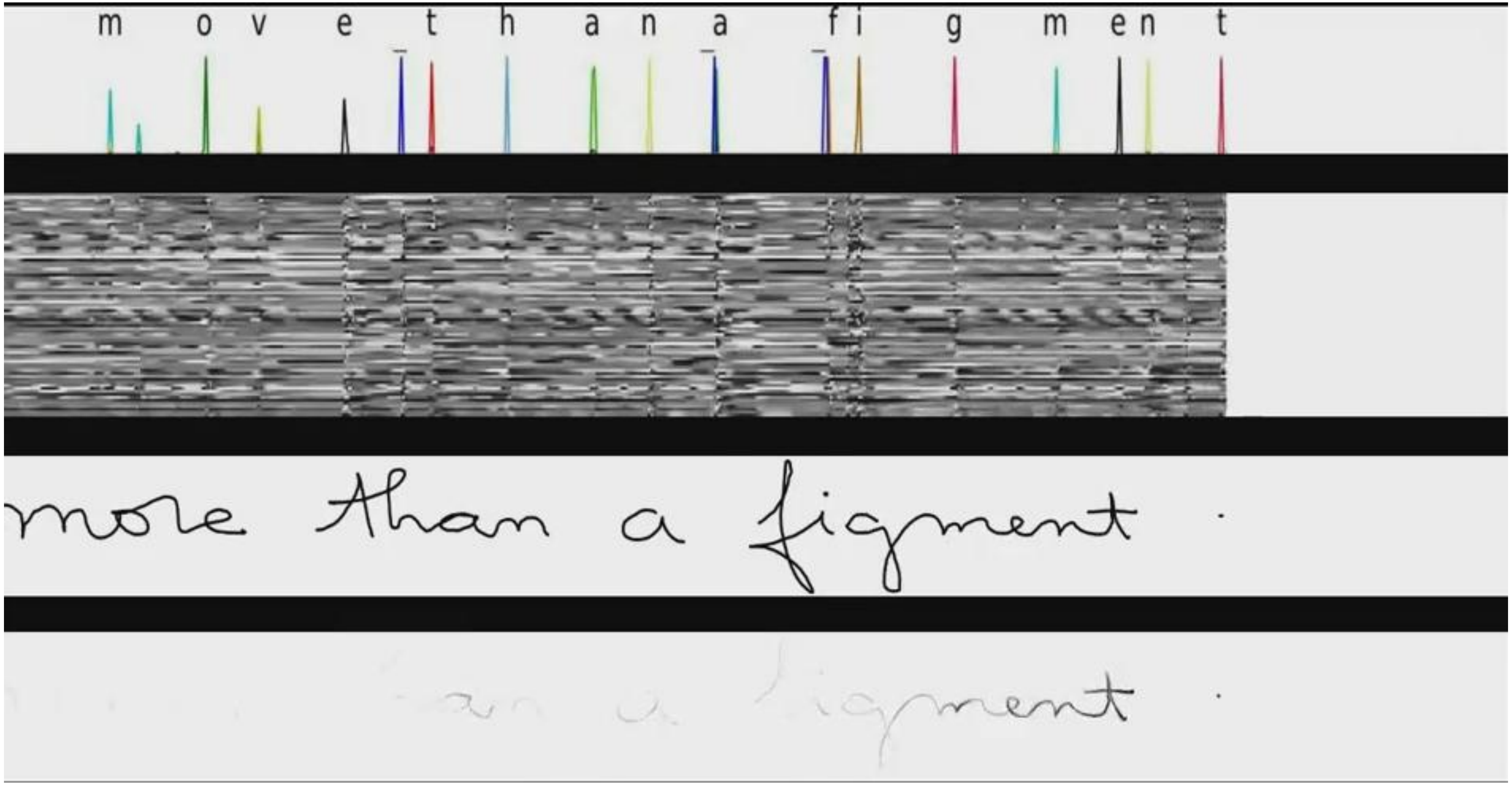


3 Connectionist Temporal Classification

3.2 **Prefix search decoding**



4 Experiment



4 Experiment

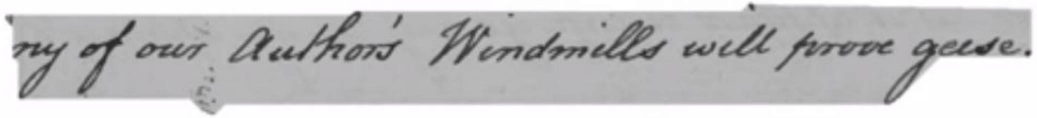
Table 1. **Label Error Rate (LER) on TIMIT.** CTC and hybrid results are means over 5 runs, ± standard error. All differences were significant ($p < 0.01$), except between weighted error BLSTM/HMM and CTC (best path).

System	LER
Context-independent HMM	38.85 %
Context-dependent HMM	35.21 %
BLSTM/HMM	33.84 ± 0.06 %
Weighted error BLSTM/HMM	31.57 ± 0.06 %
CTC (best path)	31.47 ± 0.21 %
CTC (prefix search)	30.51 ± 0.19 %

4 Experiment

HTR on Ocropus

- After 165K iterations with pretrained model



ny of our Author ' s Windmills will prove geese ."

teteee thetee theeeette ant taene faee

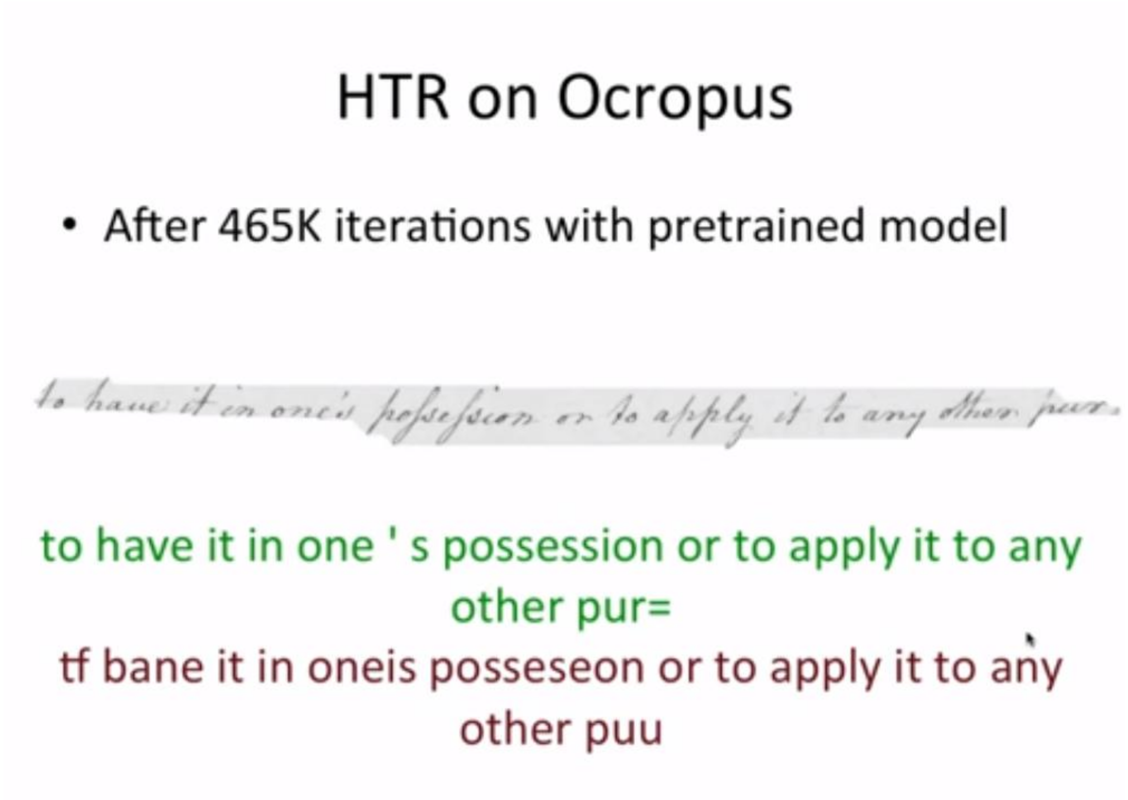
HTR on Ocropus

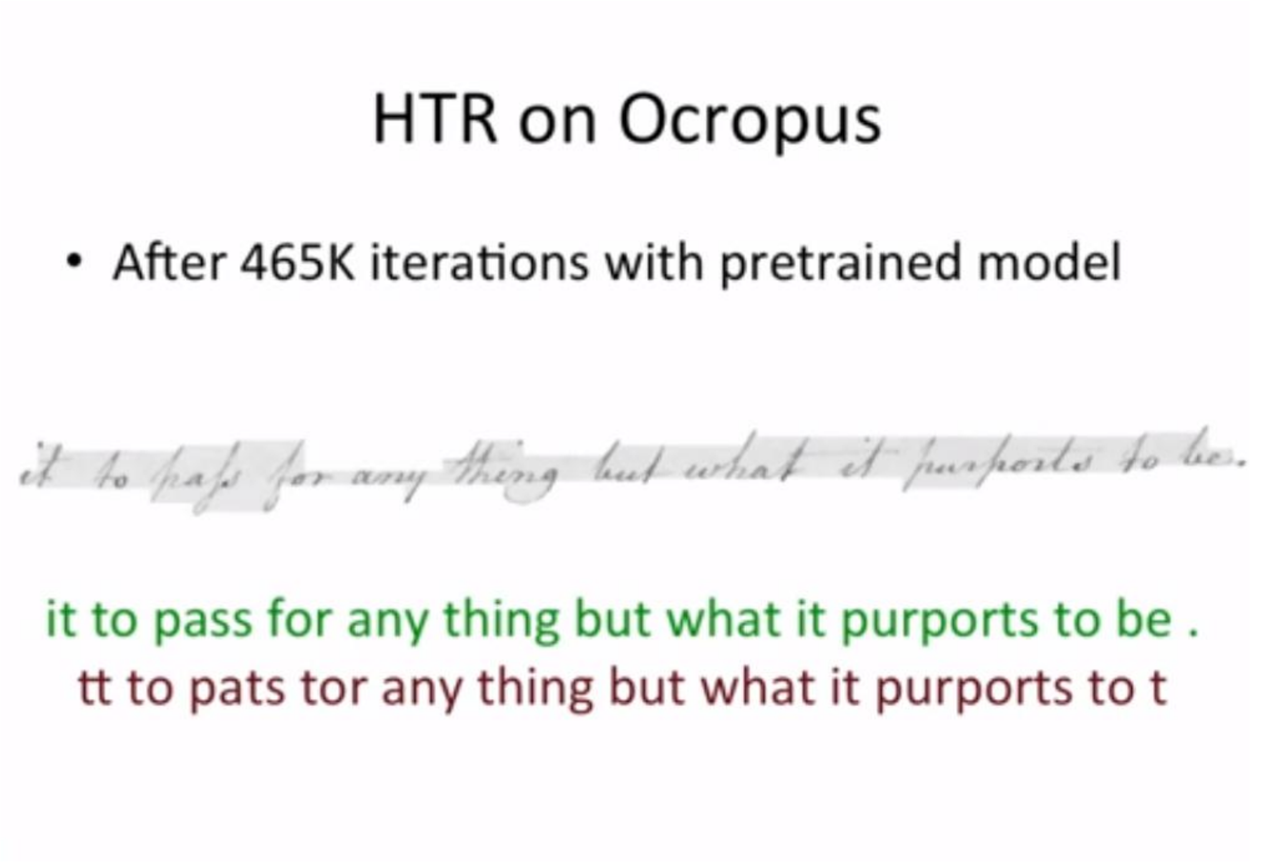
- After 165K iterations with pretrained model

15th . To present to the Court of King ' s Bench on a

aof sahaeante the fantf theg theene o ne

4 Experiment





4 Experiment

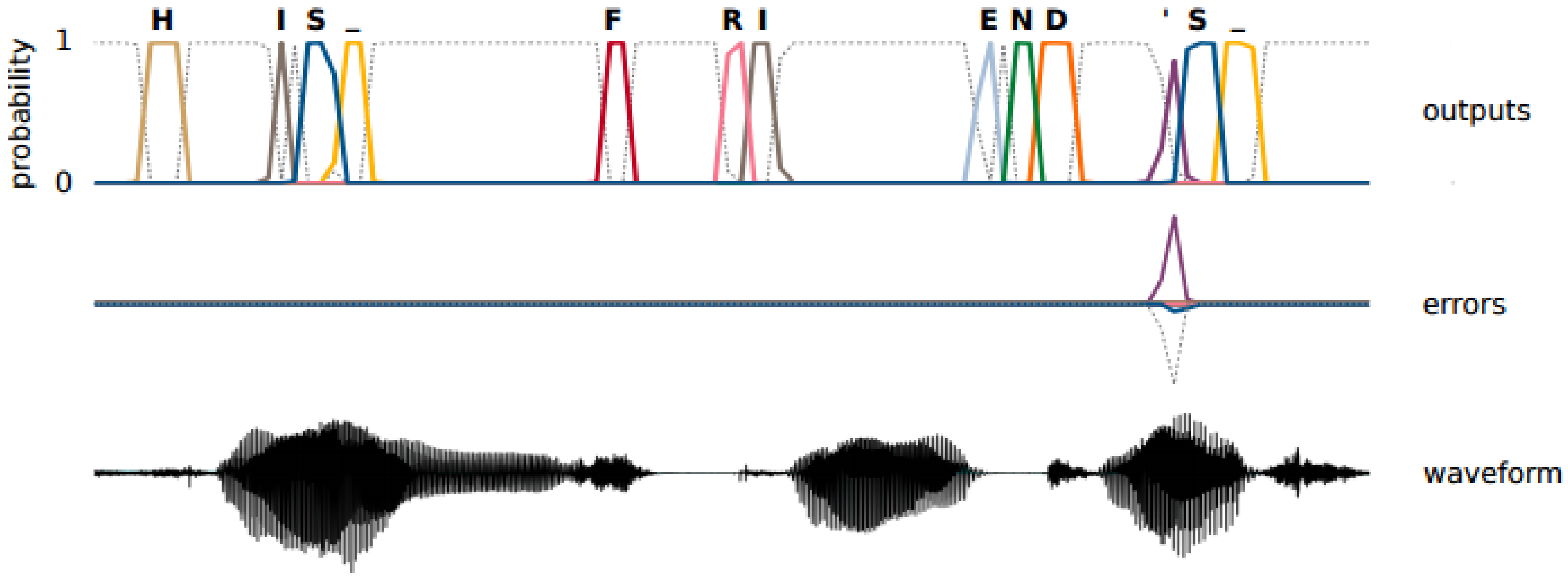


Figure 4. **Network outputs.** The figure shows the frame-level character probabilities emitted by the CTC layer (different colour for each character, dotted grey line for 'blanks'), along with the corresponding training errors, while processing an utterance. The target transcription was 'HIS_FRIENDS_', where the underscores are end-of-word markers. The network was trained with WER loss, which tends to give very sharp output decisions, and hence sparse error signals (if an output probability is 1, nothing else can be sampled, so the gradient is 0 even if the output is wrong). In this case the only gradient comes from the extraneous apostrophe before the 'S'. Note that the characters in common sequences such as 'IS', 'RI' and 'END' are emitted very close together, suggesting that the network learns them as single sounds.

4 Experiment

References

- [1] https://github.com/tmbdev/ocropy
- [2] https://github.com/junhyukoh/caffe-lstm
- [3] Graves, Alex, et al. "A novel connectionist system for unconstrained handwriting recognition." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31.5 (2009): 855-868.
- [4] Sanchez, A. Toselli, V. Romero, and E. Vidal. Icdar 2015 competition htrts: Handwritten text recognition on the transcriptorium dataset. In Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, pages 1166–1170, Aug 2015.

4 Experiment

References

Bengio., Y. (1999). Markovian models for sequential data. *Neural Computing Surveys*, *2*, 129–162.

Bishop, C. (1995). *Neural Networks for Pattern Recognition*, chapter 6. Oxford University Press, Inc.

Bourlard, H., & Morgan, N. (1994). *Connnectionist speech recognition: A hybrid approach*. Kluwer Academic Publishers.

Bridle, J. (1990). Probabilistic interpretation of feed-forward classification network outputs, with re-

[1] Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks
[2] First-Pass Large Vocabulary Continuous Speech Recognition using Bi-Directional Recurrent DNNs

Thank

You