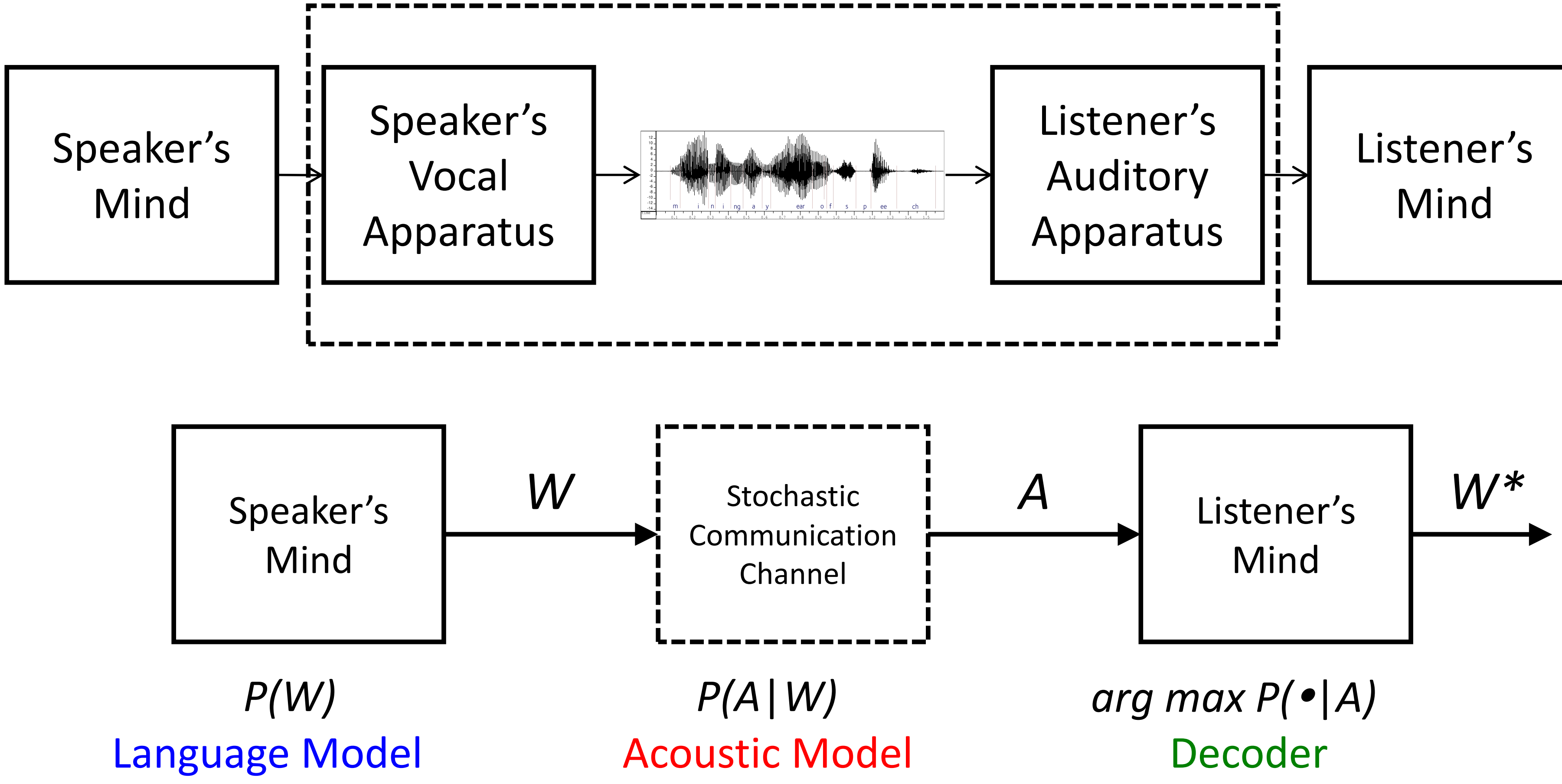
Neural Methods in Automatic Speech Recognition

IntroHLT Lecture on Nov 2, 2021

The "source-channel" model for automatic speech recognition (ASR)



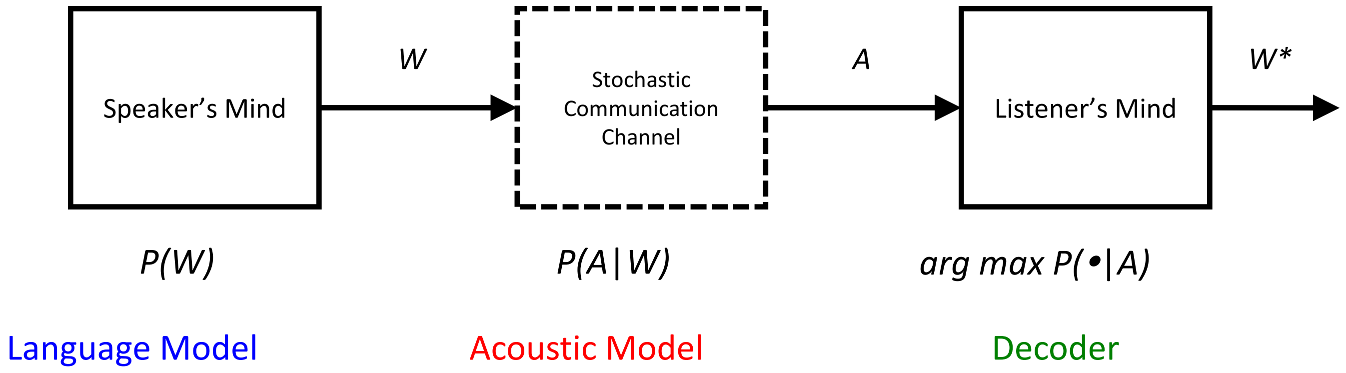
Hidden Markov models are popular as acoustic models

$$
\begin{aligned}
P(\mathbf{A} \,|\, \mathbf{W}) &= \sum_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} P(\mathbf{A}, \mathbf{S} \,|\, \mathbf{W}) = \sum_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} P(\mathbf{A} \,|\, \mathbf{S}, \mathbf{W}) P(\mathbf{S} \,|\, \mathbf{W}) \\
&\approx \sum_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} P_E(\mathbf{A} \,|\, \mathbf{S}) P_T(\mathbf{S}) \\
&= \sum_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} P_E(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_T \,|\, s_1, s_2, \ldots, s_T) P_T(s_1, s_2, \ldots, s_T) \\
&= \sum_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} \prod_{t=1}^{T} P_E(\mathbf{a}_t \,|\, s_t) P_T(s_t | s_{t-1})
\end{aligned}
$$

Dynamic programming is popular for "decoding," i.e. for hypothesis search

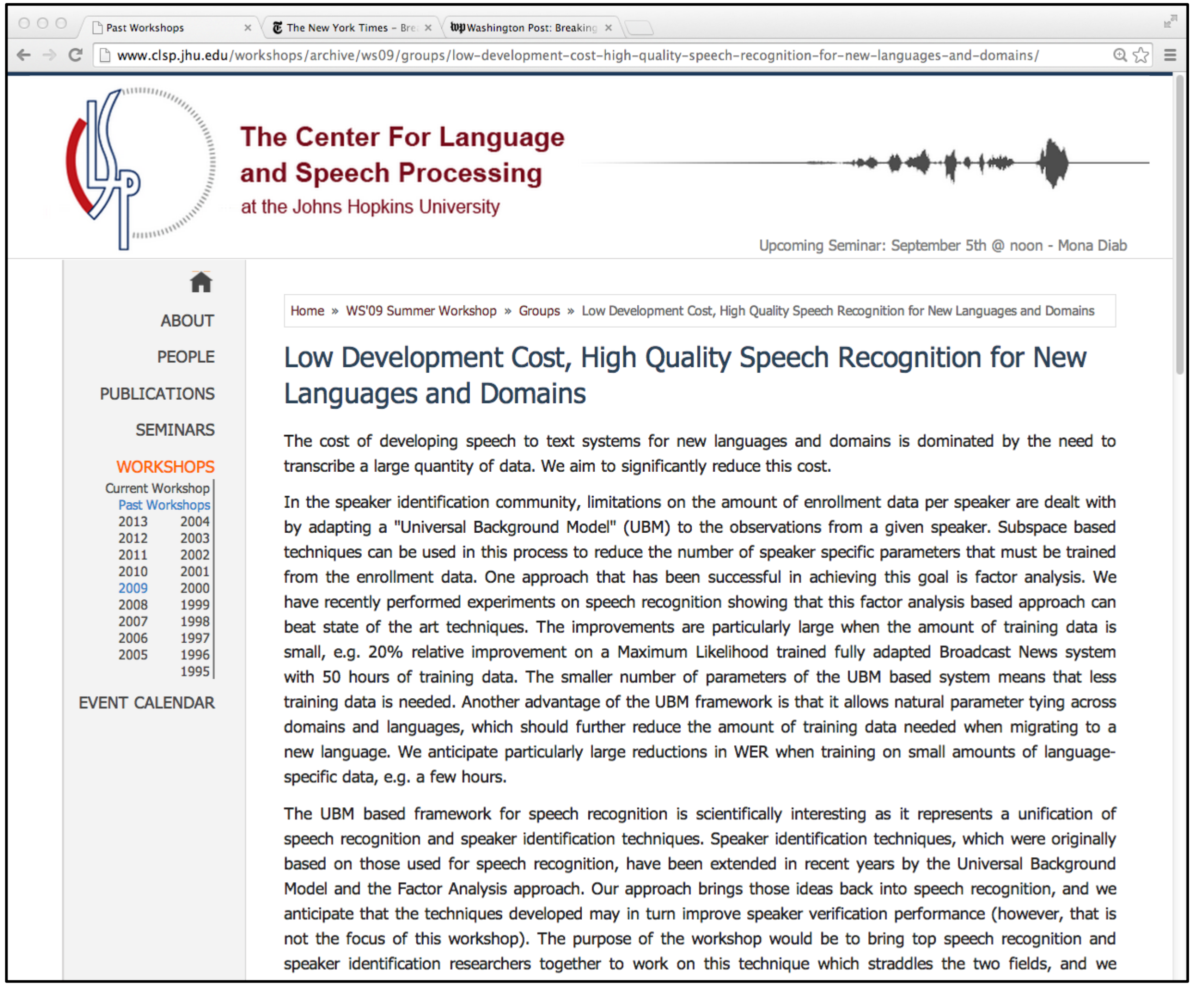
$$
\begin{aligned}
\widehat{\mathbf{W}} &= \arg\max_{\mathbf{W}} P(\mathbf{A} \,|\, \mathbf{W}) P(\mathbf{W}) \\
&= \arg\max_{\mathbf{W}} \sum_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} P(\mathbf{A} \,|\, \mathbf{S}) P(\mathbf{S}) P(\mathbf{W}) \\
&\approx \arg\max_{\mathbf{W}} \max_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} P(\mathbf{A} \,|\, \mathbf{S}) P(\mathbf{S}) P(\mathbf{W}) \\
&= \arg\max_{\mathbf{W}} \max_{\mathbf{S} \in \mathcal{S}(\mathbf{W})} \log P(\mathbf{A} \,|\, \mathbf{S}) + \log P(\mathbf{S}) + \log P(\mathbf{W}) \\
&\equiv \texttt{Project}\left(\texttt{Bestpath}\left(\texttt{Compose}\left(\mathbf{A}_{\log P(\mathbf{A} \,|\, \mathbf{S})} \circ \mathbf{L}_{\log P(\mathbf{S})} \circ \mathbf{G}_{\log P(\mathbf{W})}\right)\right)\right)
\end{aligned}
$$

The ASR Landscape in 2009



- Commercial providers had proprietary algorithms and software
- Academic software tools were mostly good only for research
  - Usually not scalable for deployment
  - Often required licensing for commercial use
- Significant barriers to entry existed for start-ups and small(er) labs
  - Algorithms were complex to understand and implement
  - Significant "black art" beyond the algorithms themselves was needed

Kaldi was born in the Summer of 2009



Past Workshops | The New York Times – Bre… | Washington Post: Breaking…

www.clsp.jhu.edu/workshops/archive/ws09/groups/low-development-cost-high-quality-speech-recognition-for-new-languages-and-domains/

The Center For Language and Speech Processing
at the Johns Hopkins University

Upcoming Seminar: September 5th @ noon - Mona Diab

ABOUT
PEOPLE
PUBLICATIONS
SEMINARS
WORKSHOPS
Current Workshop
Past Workshops
2013 2012 2011 2010 2009 2008 2007 2006 2005
2004 2003 2002 2001 2000 1999 1998 1997 1996 1995
EVENT CALENDAR

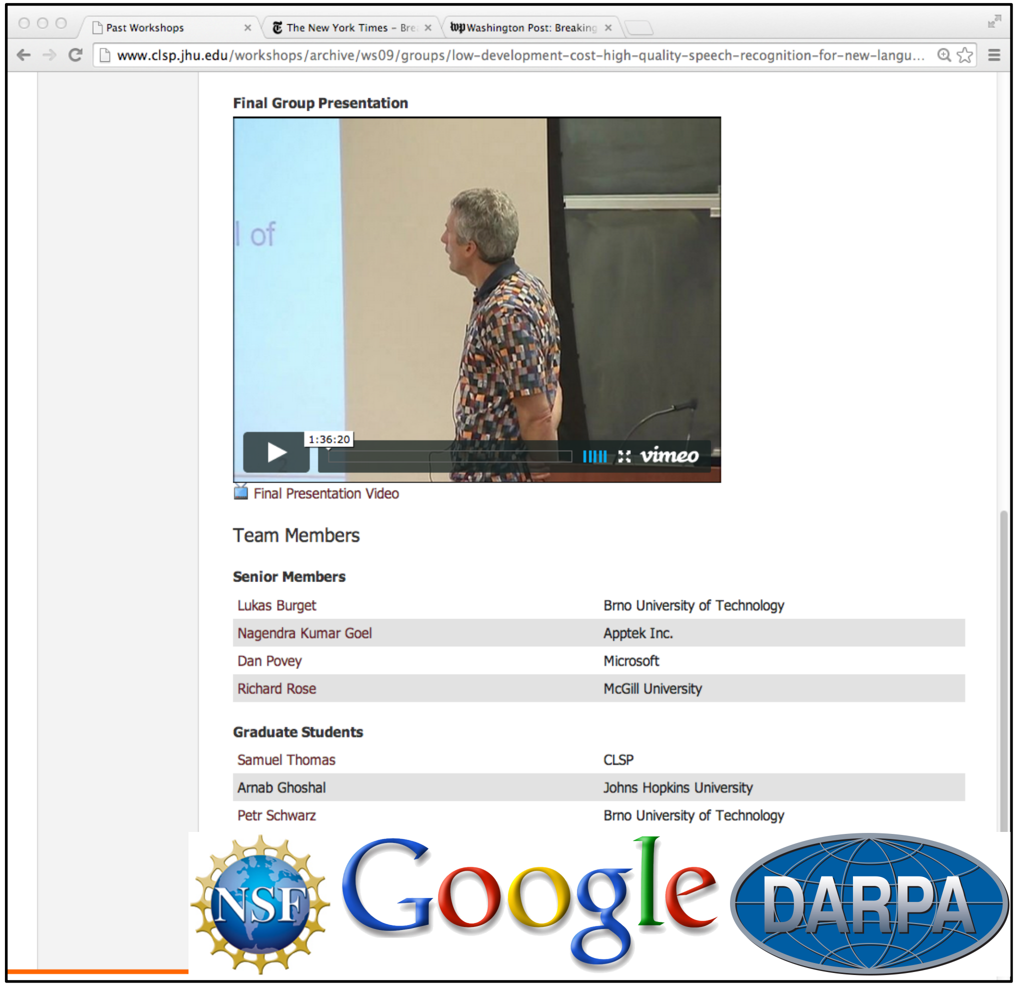
Home » WS'09 Summer Workshop » Groups » Low Development Cost, High Quality Speech Recognition for New Languages and Domains

Low Development Cost, High Quality Speech Recognition for New Languages and Domains

The cost of developing speech to text systems for new languages and domains is dominated by the need to transcribe a large quantity of data. We aim to significantly reduce this cost.

In the speaker identification community, limitations on the amount of enrollment data per speaker are dealt with by adapting a "Universal Background Model" (UBM) to the observations from a given speaker. Subspace based techniques can be used in this process to reduce the number of speaker specific parameters that must be trained from the enrollment data. One approach that has been successful in achieving this goal is factor analysis. We have recently performed experiments on speech recognition showing that this factor analysis based approach can beat state of the art techniques. The improvements are particularly large when the amount of training data is small, e.g. 20% relative improvement on a Maximum Likelihood trained fully adapted Broadcast News system with 50 hours of training data. The smaller number of parameters of the UBM based system means that less training data is needed. Another advantage of the UBM framework is that it allows natural parameter tying across domains and languages, which should further reduce the amount of training data needed when migrating to a new language. We anticipate particularly large reductions in WER when training on small amounts of language-specific data, e.g. a few hours.

The UBM based framework for speech recognition is scientifically interesting as it represents a unification of speech recognition and speaker identification techniques. Speaker identification techniques, which were originally based on those used for speech recognition, have been extended in recent years by the Universal Background Model and the Factor Analysis approach. Our approach brings those ideas back into speech recognition, and we anticipate that the techniques developed may in turn improve speaker verification performance (however, that is not the focus of this workshop). The purpose of the workshop would be to bring top speech recognition and speaker identification researchers together to work on this technique which straddles the two fields, and we



Past Workshops | The New York Times – Bre… | Washington Post: Breaking…

www.clsp.jhu.edu/workshops/archive/ws09/groups/low-development-cost-high-quality-speech-recognition-for-new-langu…

Final Group Presentation

1:36:20 vimeo

Final Presentation Video

Team Members

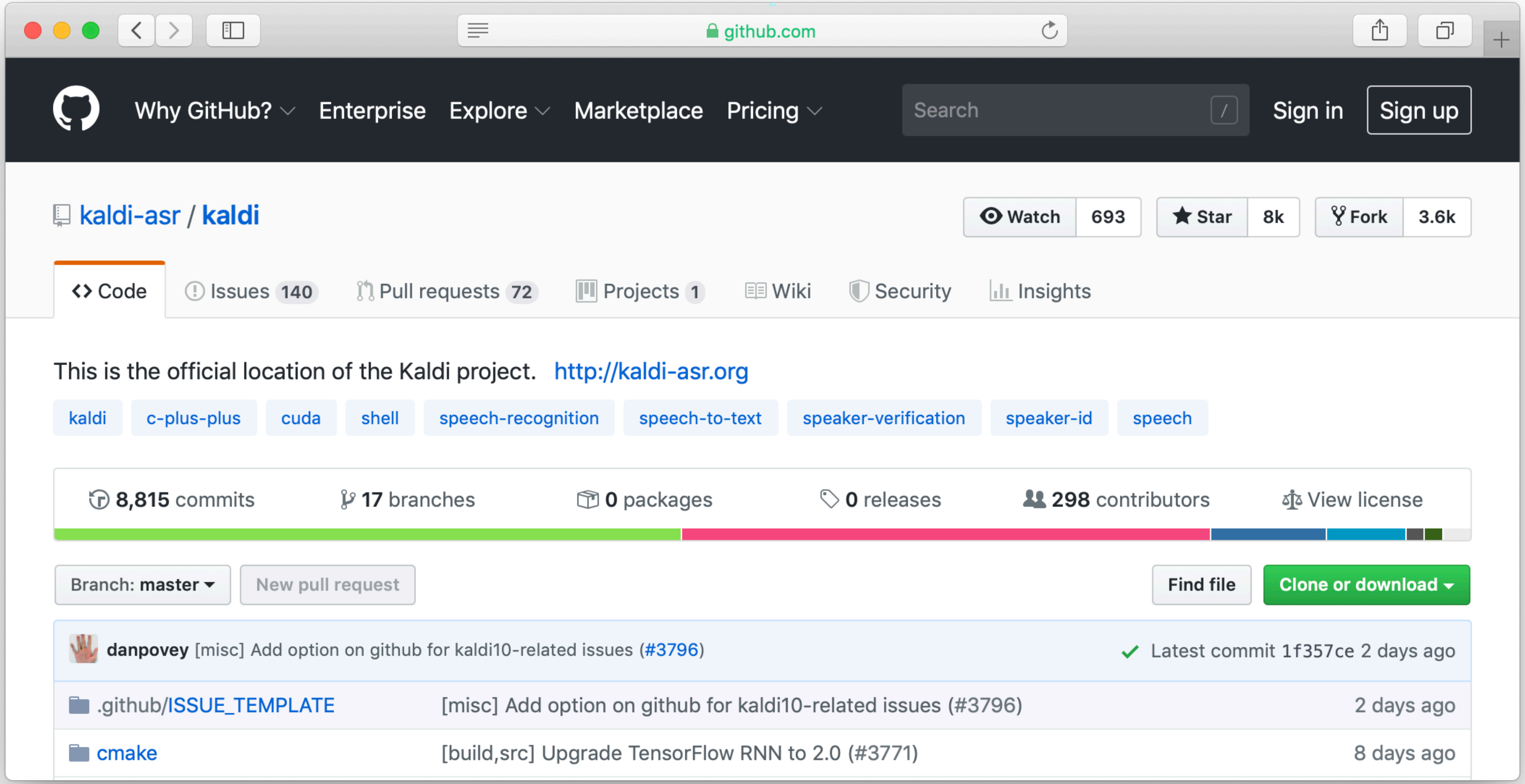
Senior Members

Lukas Burget	Brno University of Technology
Nagendra Kumar Goel	Apptek Inc.
Dan Povey	Microsoft
Richard Rose	McGill University

Graduate Students

Samuel Thomas	CLSP
Arnab Ghoshal	Johns Hopkins University
Petr Schwarz	Brno University of Technology

Kaldi: Legendary Ethiopian goatherd who discovered coffee



Kaldi today: A community of researchers cooperatively advancing ASR

- C++ library, command line tools, several ASR recipes
  - Freely available via GitHub (Apache 2.0 license)
- Top performance in open benchmark tests
  - NIST OpenKWS 2014, IARPA ASpIRE 2015, NIST LoReHLT 2018, …, MUCS 2021
- Widely adopted in academia and extensively used in industry
  - 300+ citations in 2014 (based on Google Scholar data)
  - 400+ citations in 2015, 600+ citations in 2016, 800+ citations in 2017, …
  - Used (& developed further) by several US and non-US companies
- Kaldi "trunk" maintained by Dan Povey @ xiaomi and Jan Trmal @ jhu
  - Forks contain specializations by others (including other Hopkins researchers)

Staying ahead of the field: 2012-Today

- ASR technology is advancing very rapidly
  - Amazon, Apple, Baidu, Facebook, Google, Microsoft, Tencent, …
- Kaldi leads the field with innovations, big and small, …
  - From SGMMs to DNNs (2012)
  - From English to “low resource” languages (2013, IARPA BABEL)
  - Parallelization of DNN training (2014, Natural Gradient SGD)
  - From close-talking to far-field recordings (2015, IARPA ASpIRE)
  - Chain models: better, cheaper and faster (2016)
  - Backstitch: adversarial training reinterpreted (2017)
  - TDNN-F acoustic models (2018)
  - GPU acceleration of Viterbi decoding (2019)
- … and tries to keep up with advances made by others

A paper appeared in September 2011 …

INTERSPEECH 2011

Conversational Speech Transcription
Using Context-Dependent Deep Neural Networks

Frank Seide[1], *Gang Li,*[1] *and Dong Yu*[2]

[1]Microsoft Research Asia, Beijing, P.R.C.
[2]Microsoft Research, Redmond, USA
{fseide,g

ICASSP 1988

Phoneme Recognition: Neural Networks vs.
Hidden Markov Models

A. Waibel T. Hanazawa G. Hinton * K. Shikano K. Lang †

ATR Interpreting Telephony Research Laboratories
*University of Toronto and Canadian Institute for Advanced Research
†Carnegie-Mellon University

So, ~~a lot of~~ progress has been made since 1988

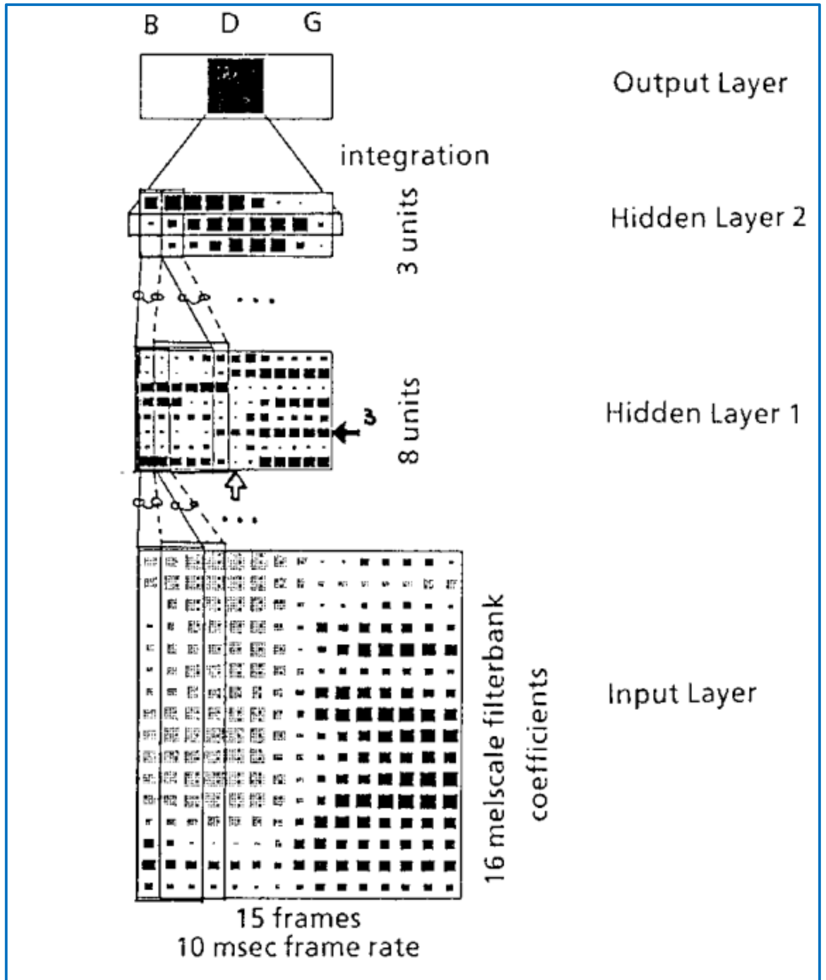
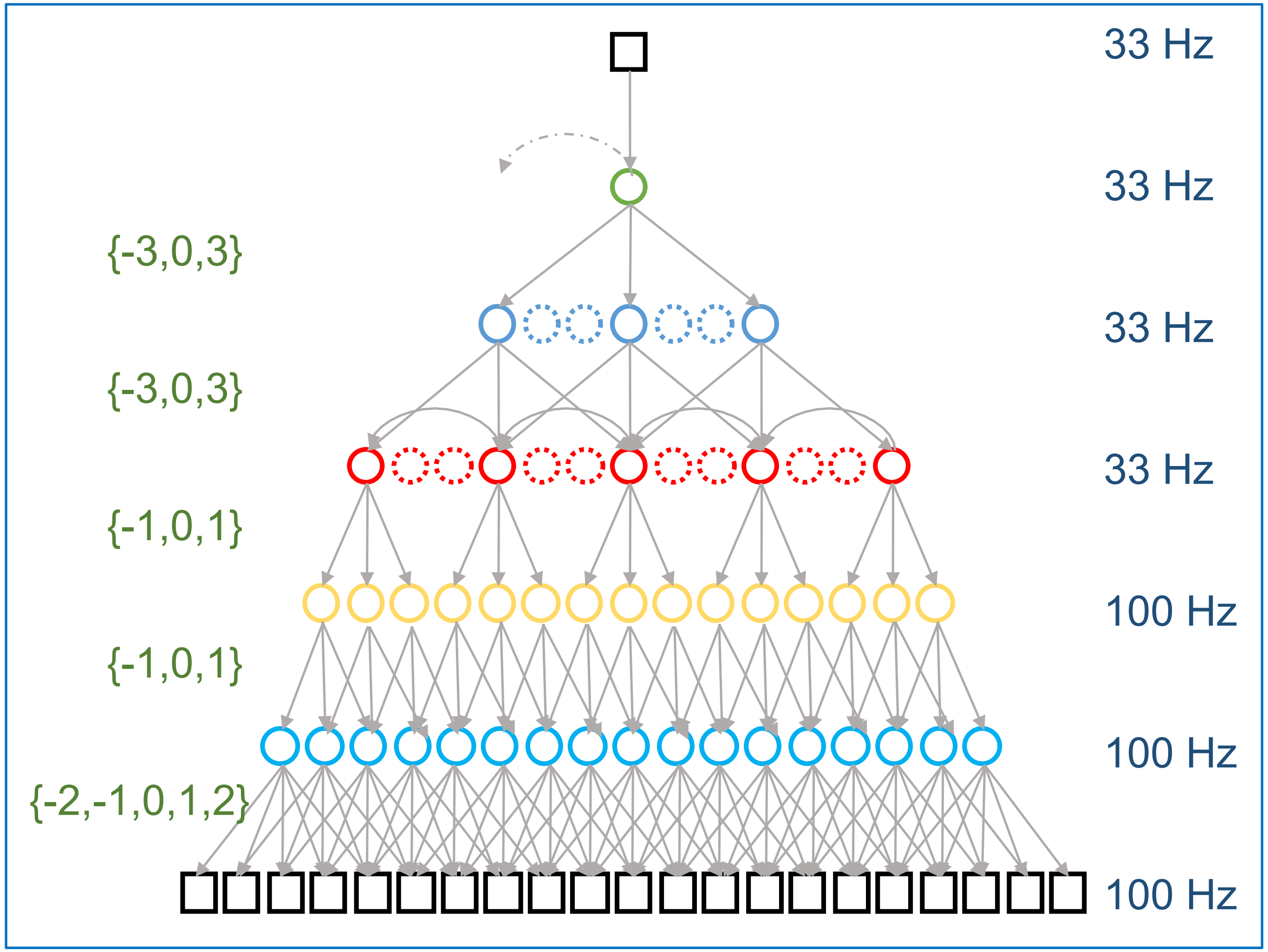


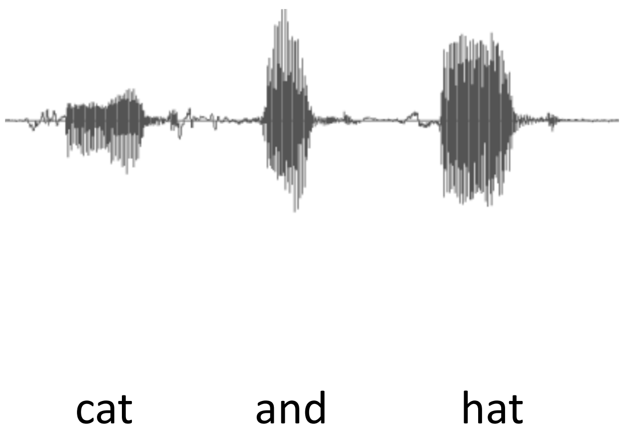
Figure 2: The TDNN architecture (input: "DA")



Acoustic Modeling with Deep Neural Networks for Hybrid ASR Systems

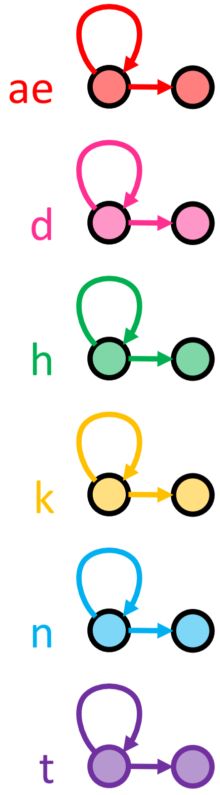
Repurposing Algorithms Developed for HMM-based Architectures

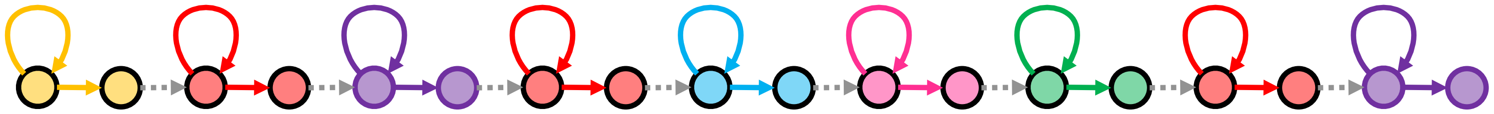
Composite HMM for "cat and hat"

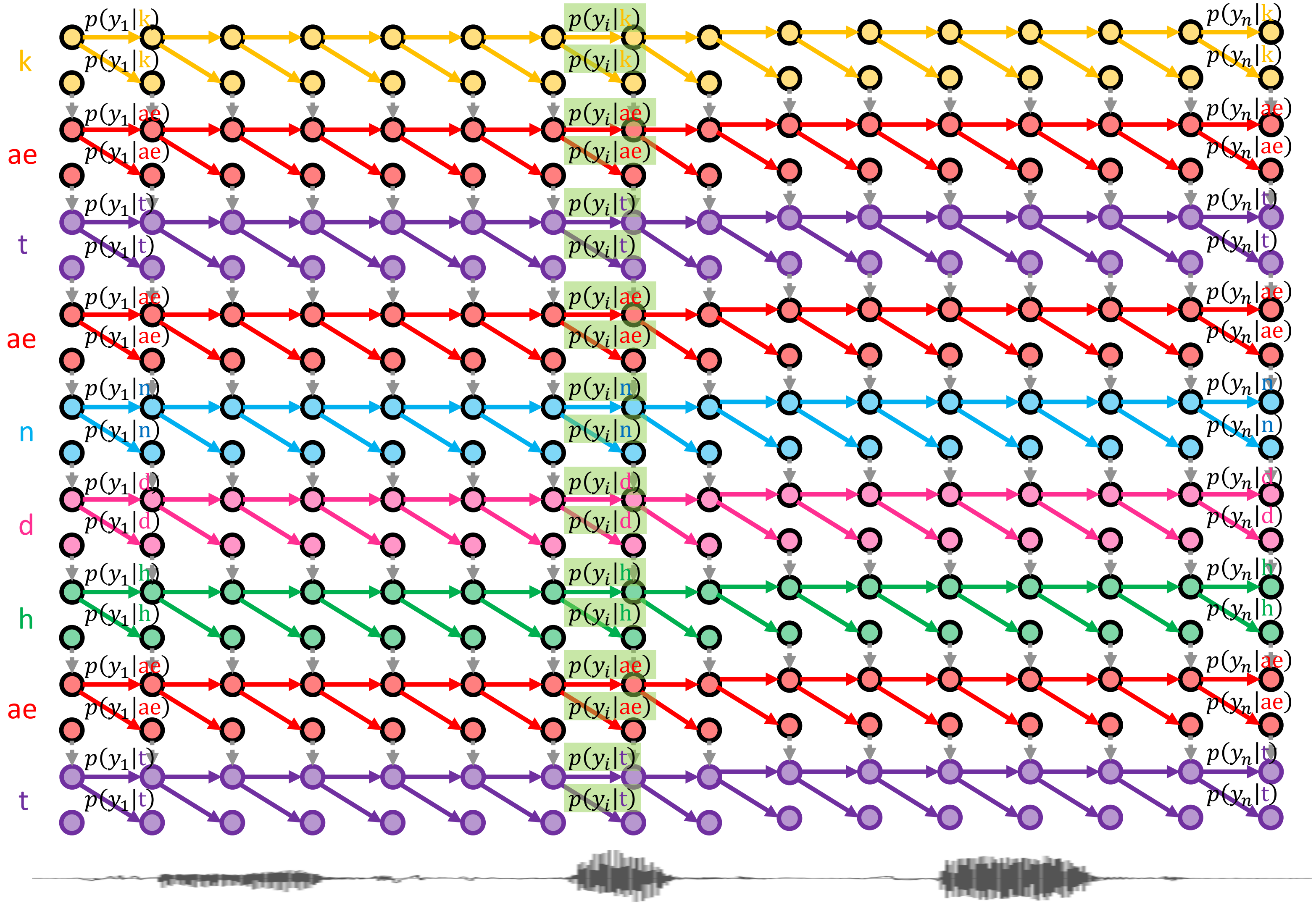


and	ae n d
cat	k ae t
hat	h ae t

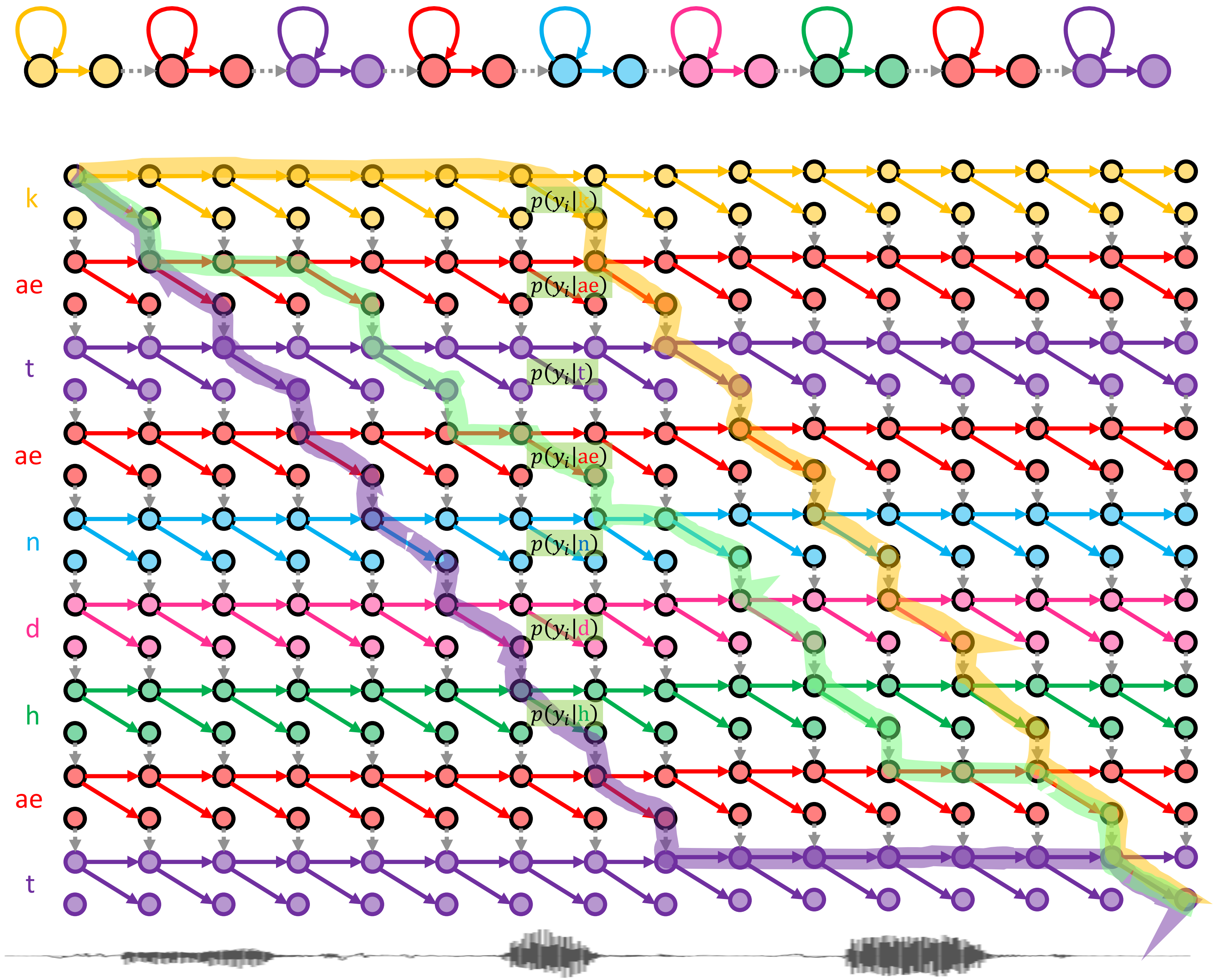
Phoneme HMMs







Composite HMM for “cat and hat”



“Forward” Algorithm

$$P(\boldsymbol{y}|\boldsymbol{w}) = \sum_{\boldsymbol{s}\in\mathcal{S}(\boldsymbol{w})} P_{\vartheta}(\boldsymbol{y}|\boldsymbol{s})P_{\tau}(\boldsymbol{s})$$

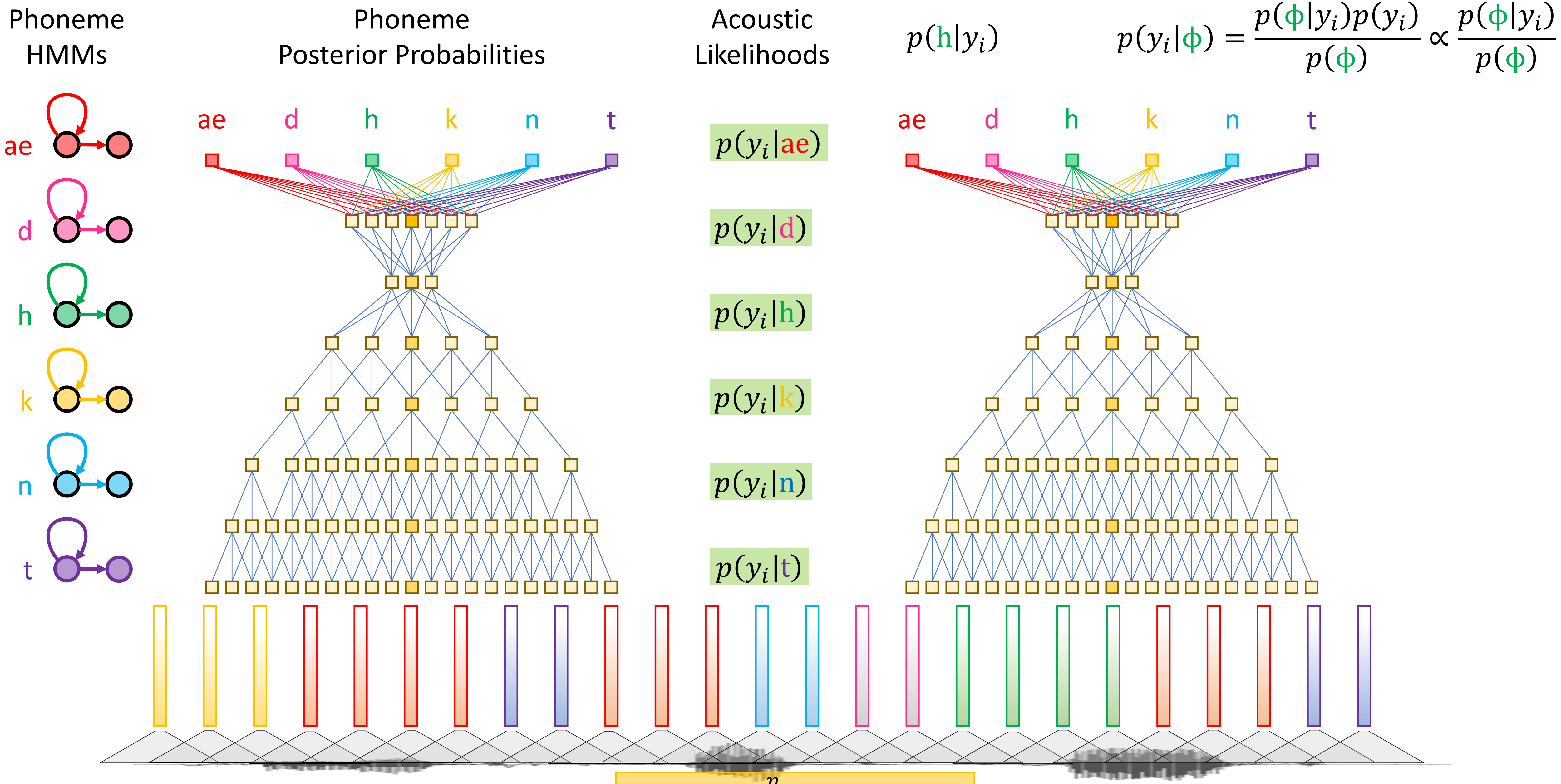
$$= \sum_{\boldsymbol{s}\in\mathcal{S}(\boldsymbol{w})} \prod_{i=1}^{n} P_{\vartheta}(y_i|s_i)P_{\tau}(s_i|s_{i-1})$$

Viterbi Algorithm

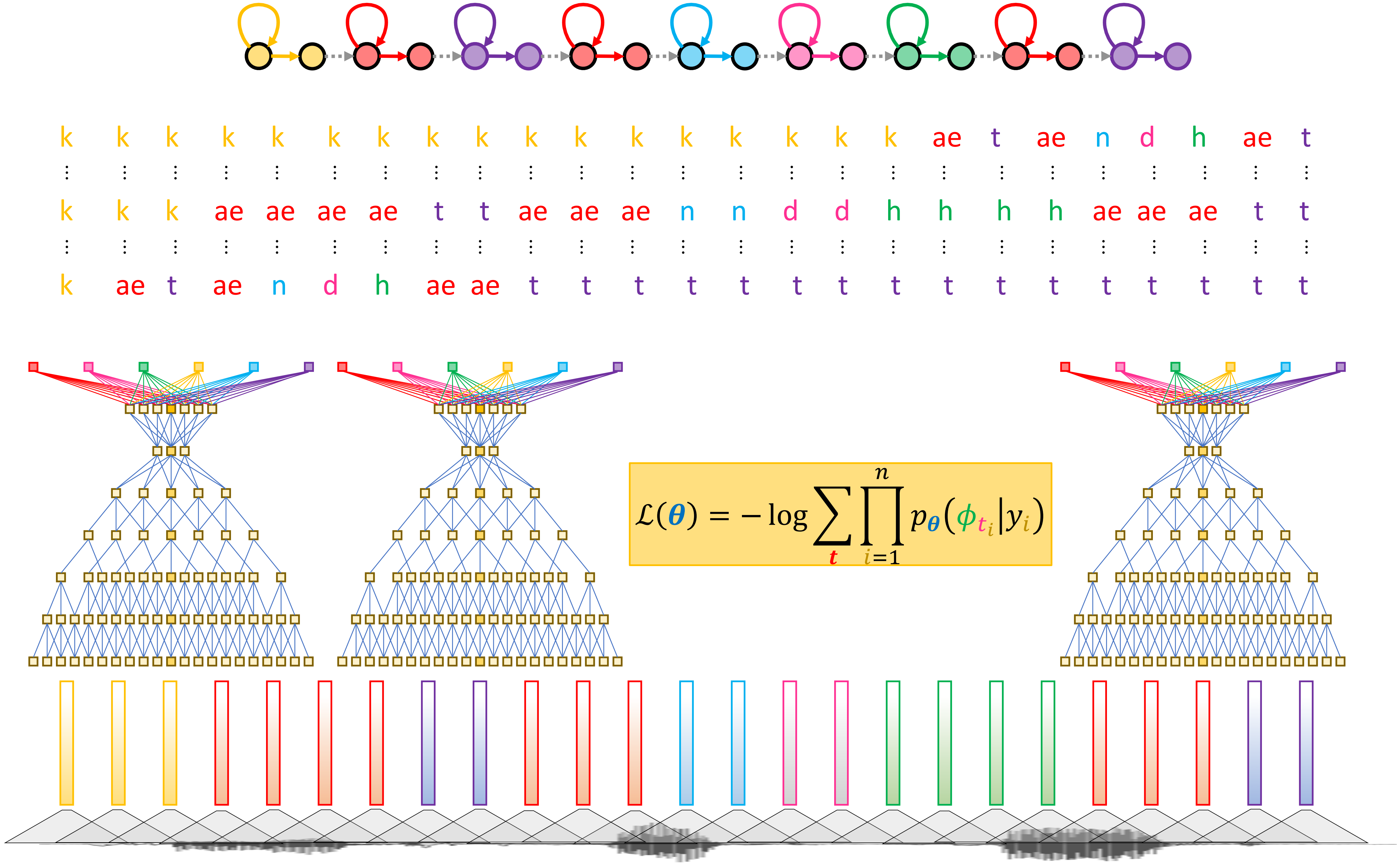
$$\hat{\boldsymbol{s}} = \arg\max_{\boldsymbol{s}\in\mathcal{S}(\boldsymbol{w})} P(\boldsymbol{s}|\boldsymbol{y})$$

$$= \arg\max_{\boldsymbol{s}\in\mathcal{S}(\boldsymbol{w})} \frac{P(\boldsymbol{y},\boldsymbol{s})}{P(\boldsymbol{y})}$$

$$= \arg\max_{\boldsymbol{s}\in\mathcal{S}(\boldsymbol{w})} \prod_{i=1}^{n} P_{\vartheta}(y_i|s_i)P_{\tau}(s_i|s_{i-1})$$



$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{i=1}^{n} \log p_{\theta}(\hat{\phi}_i|y_i)$$



k k k k k k k k k k k k k k k k k ae t ae n d h ae t
k k k ae ae ae ae t t ae ae ae n n d d h h h h ae ae ae t t
k ae t ae n d h ae ae t t t t t t t t t t t t t t t t
$\mathcal{L}(\boldsymbol{\theta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\boldsymbol{\theta}}(\phi_{t_i} | y_i)$

Language Modeling
with
(Recurrent) Neural Networks

Efforts to Get Further Away from GMM-HMM Architectures

Using Neural Networks to Estimate $P(w_t|h_t)$

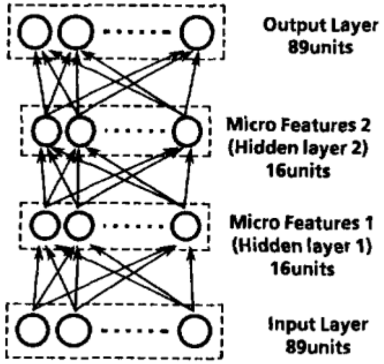


Fig.2 Basic Bigram Network for Word Category Prediction

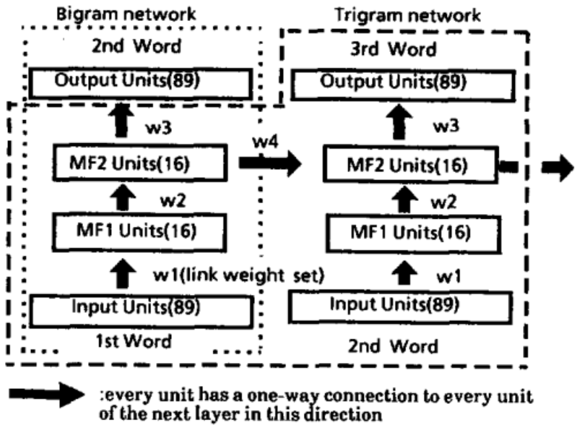


Fig.3 NETgram Model 1 for Word Category Prediction

A STUDY OF ENGLISH WORD CATEGORY PREDICTION BASED ON NEURAL NETWORKS

Masami NAKAMURA, Kiyohiro SHIKANO

ATR Interpreting Telephony Research Laboratories
Seika-chou, Souraku-gun, Kyoto 619-02, JAPAN

ICASSP 1989

ELSEVIER

Available online at www.sciencedirect

ScienceDirect

Computer Speech and Language 21 (2007) 492–518

LANGUAGE

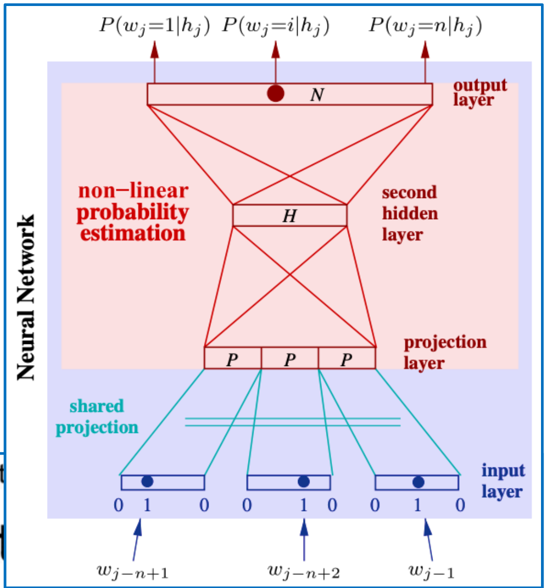
www.elsevier.com/locate/csl

Continuous space language models ☆

Holger Schwenk

Spoken Language Processing Group, LIMSI-CNRS, BP 133, 91403 Orsay cedex, France

Received 19 December 2005; received in revised form 15 September 2006; accepted 15 September 2006
Available online 9 October 2006



A paper appeared in September 2010 …

COGNITIVE SCIENCE **14**, 179–211 (1990)

Finding Structure in Time

JEFFREY L. ELMAN
University of California, San Diego

INTERSPEECH 2010

Recurrent neural network based language model

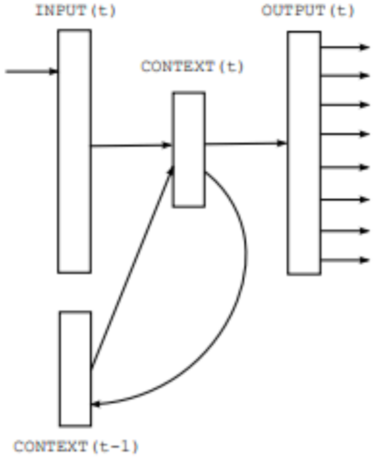
Tomáš Mikolov[1,2], Martin Karafiát[1], Lukáš Burget[1], Jan "Honza" Černocký[1], Sanjeev Khudanpur[2]

[1]Speech@FIT, Brno University of Technology, Czech Republic
[2] Department of Electrical and Computer Engineering, Johns Hopkins University, USA
{imikolov,karafiat,burget,cernocky}@fit.vutbr.cz, khudanpur@jhu.edu

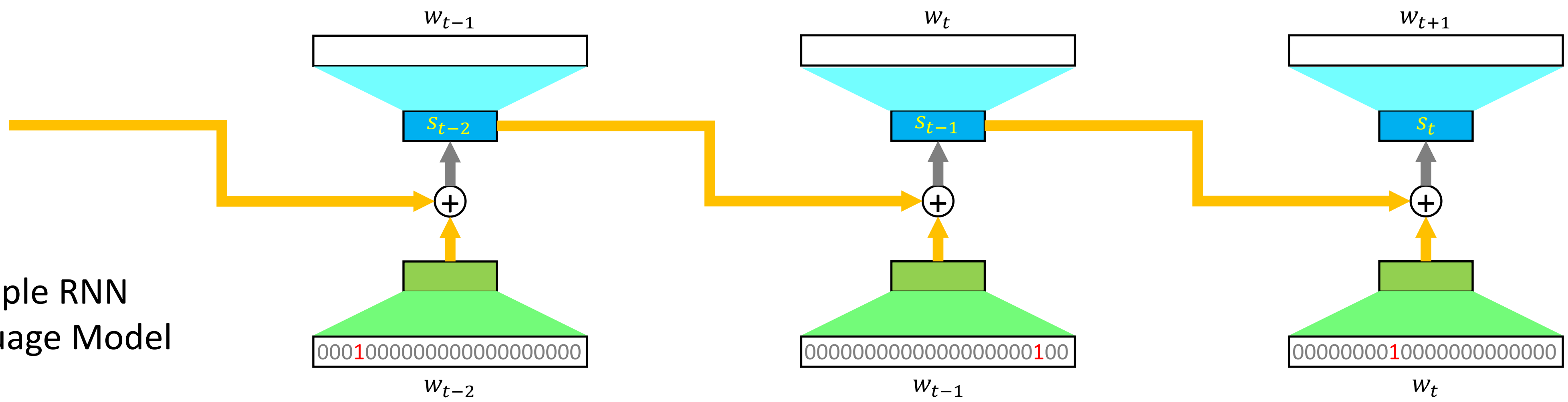
Abstract

A new recurrent neural network based language model (RNN LM) with applications to speech recognition is presented. Results indicate that it is possible to obtain around 50% reduction of perplexity by using mixture of several RNN LMs, compared to a state of the art backoff language model. Speech recognition experiments show around 18% reduction of word error rate on the Wall Street Journal task when comparing models trained on the same amount of data, and around 5% on the much harder NIST RT05 task, even when the backoff model is trained on much more data than the RNN LM. We provide ample empirical evidence to suggest that connectionist language models are superior to standard n-gram techniques, except their high computational (training) complexity.

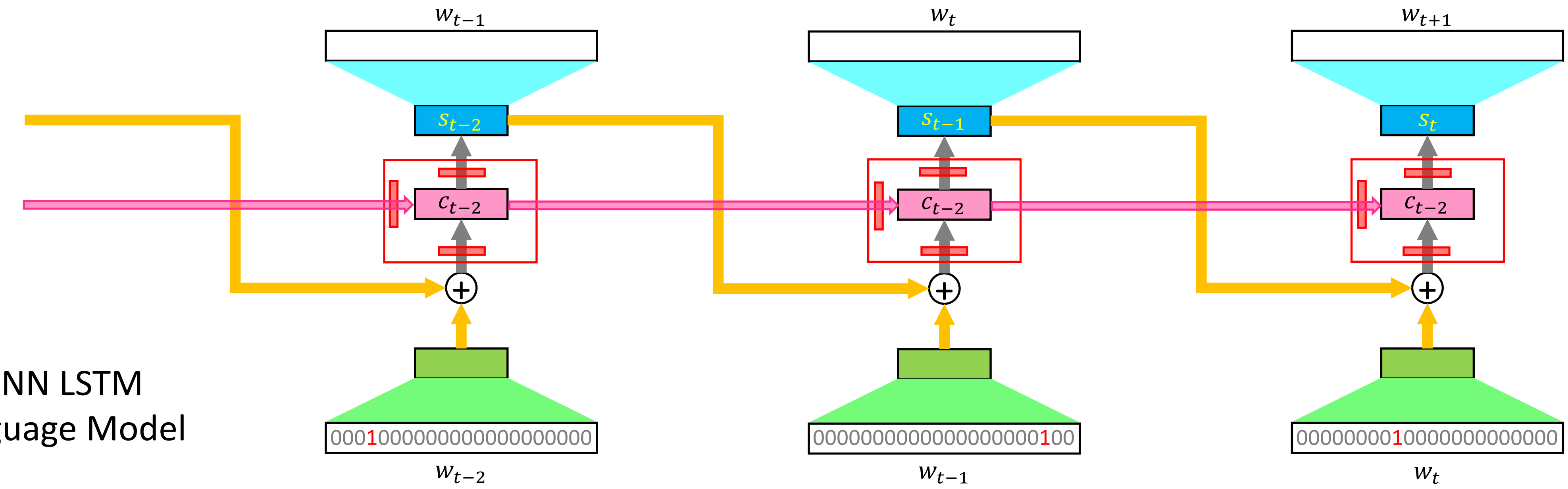
Index Terms: language modeling, recurrent neural networks, speech recognition



A Simple RNN
Language Model



An RNN LSTM
Language Model



Speech Recognition without the HMM “Backend”

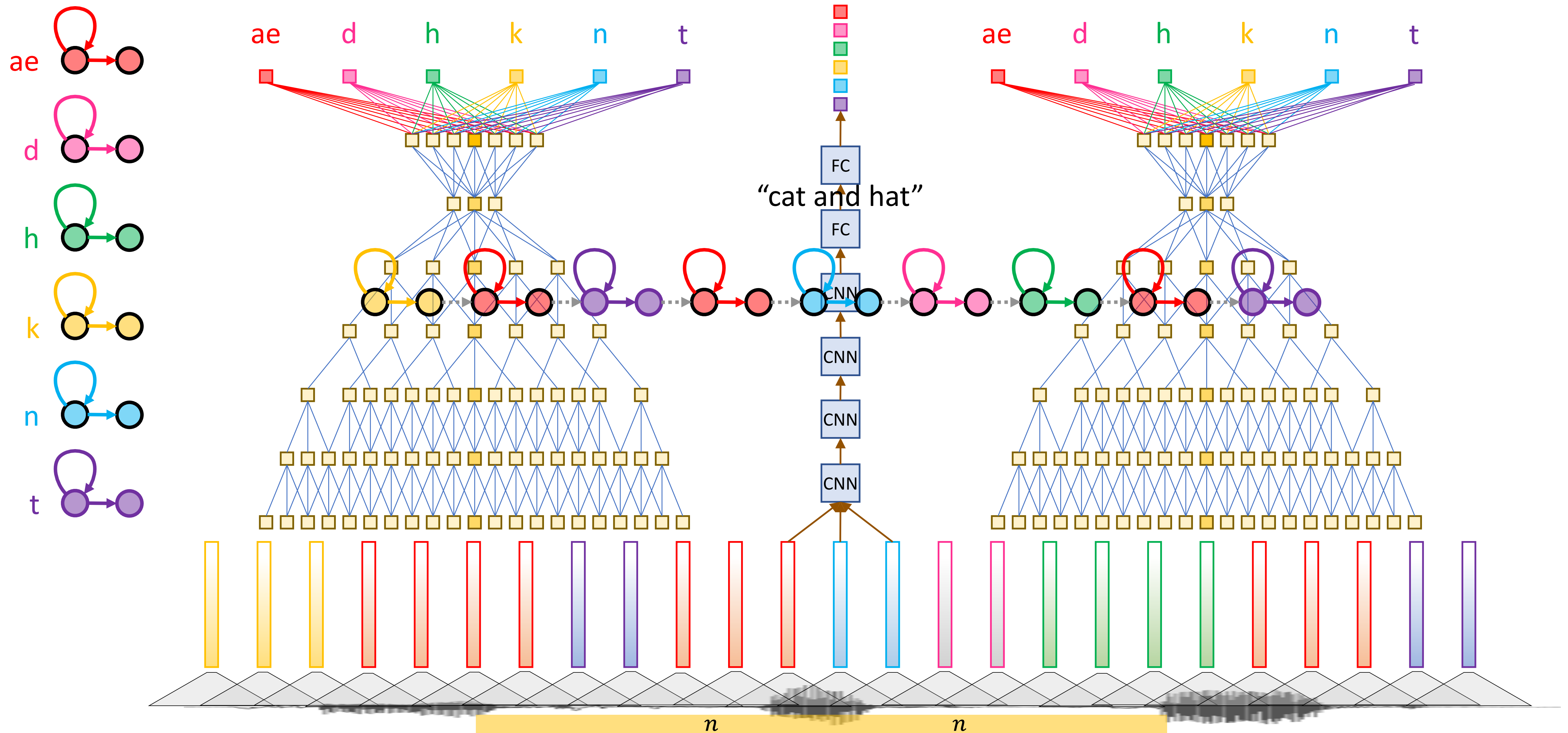
Efforts to Get Away from GMM-HMM Architectures

Phoneme HMMs

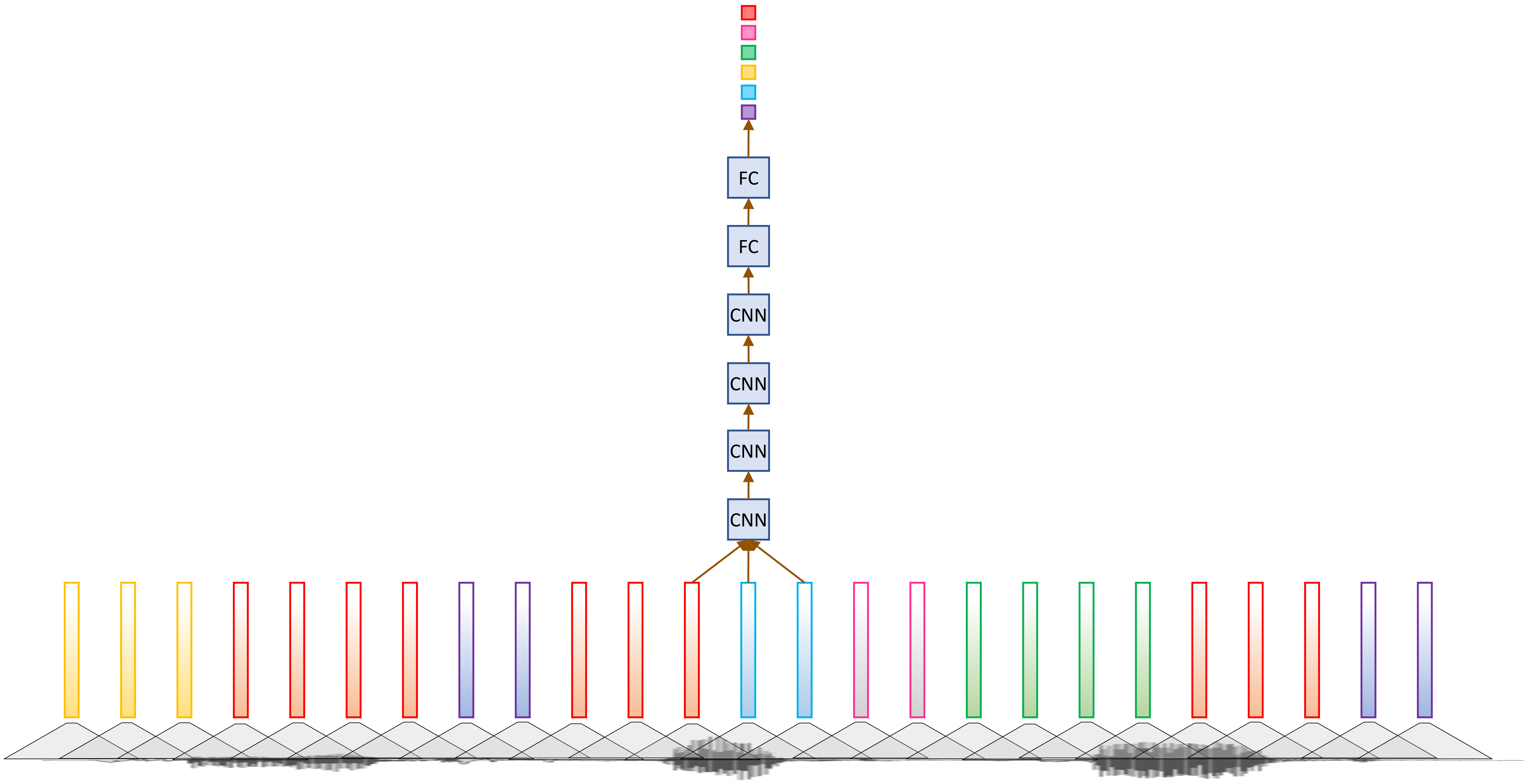
Phoneme Posterior Probabilities

$p(\phi|y_i)$

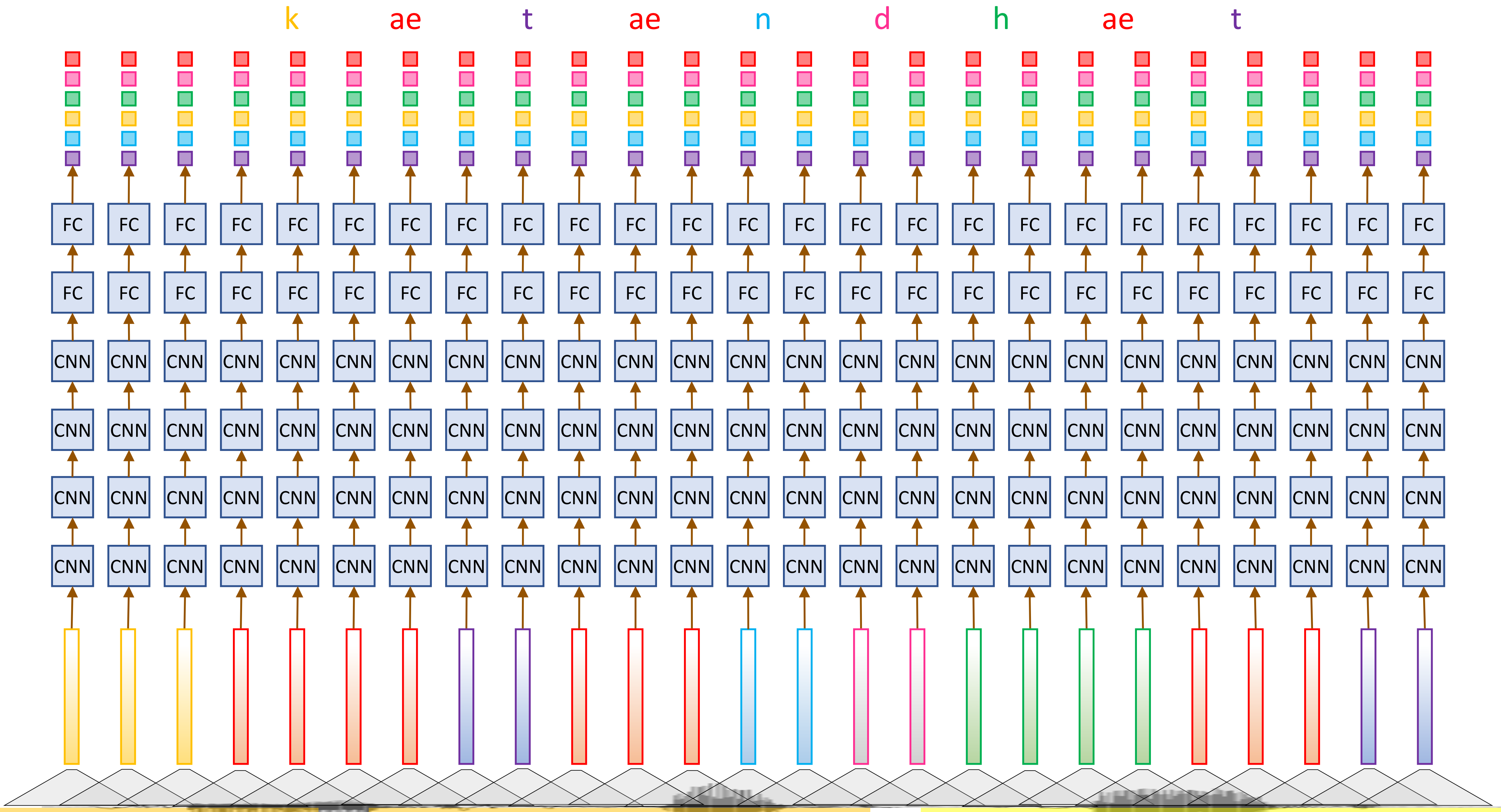
$$p(y_i|\phi) = \frac{p(\phi|y_i)p(y_i)}{p(\phi)} \propto \frac{p(\phi|y_i)}{p(\phi)}$$



$$\mathcal{L}_{\mathrm{CE}}(\boldsymbol{\theta}) = -\log \prod_{i=1}^{n} p_\theta(\hat{\phi}_i|y_i) = -\sum_{i=1}^{n} \log p_\theta(\hat{\phi}_i|y_i)$$



FC
FC
CNN
CNN
CNN
CNN



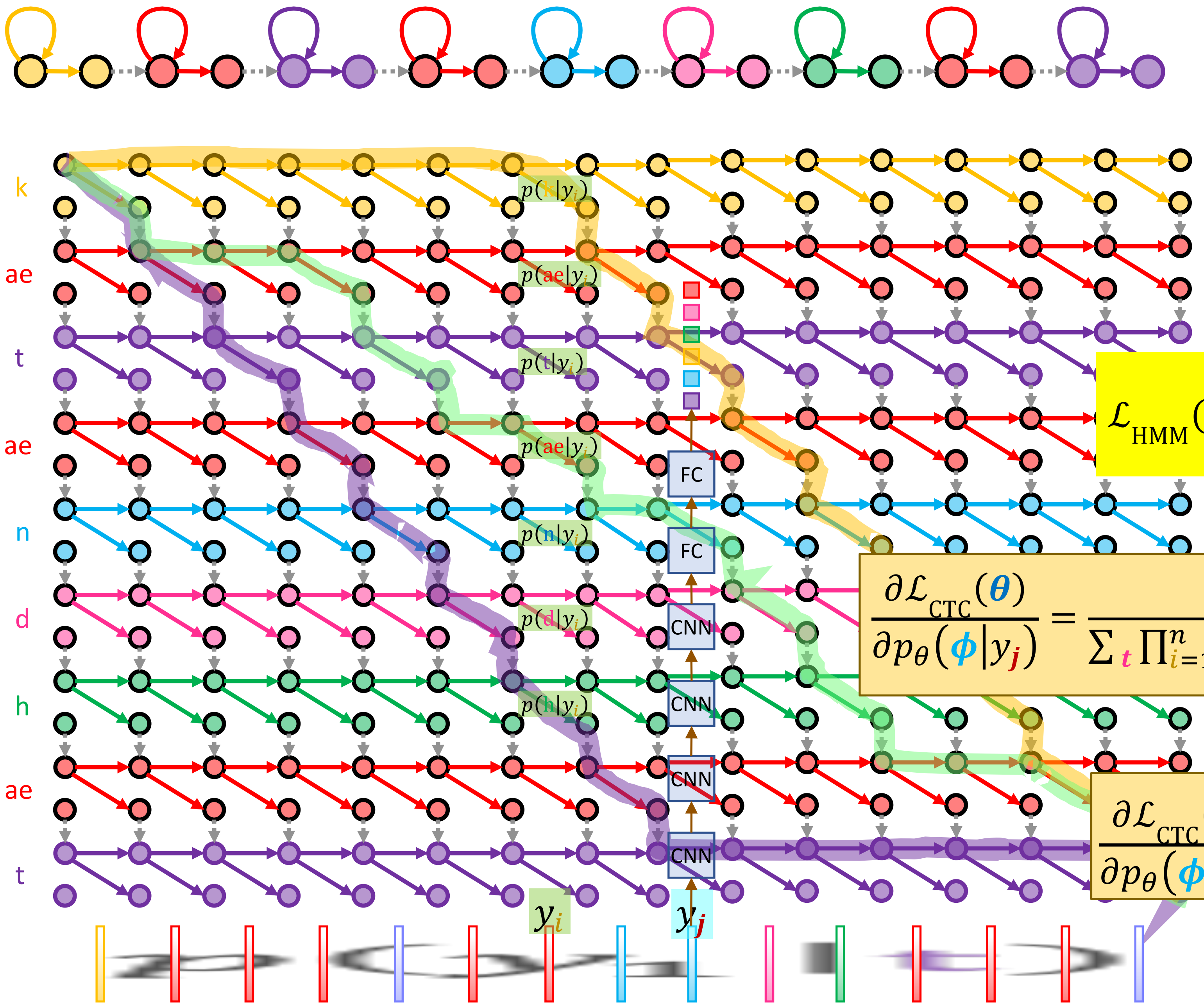
$$\mathcal{L}_{\text{CE}}(\boldsymbol{\theta}) = -\log \prod_{i=1}^{n} p_{\theta}(\hat{\phi}_i | y_i)$$

$$\mathcal{L}_{\text{CTC}}(\boldsymbol{\theta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i} | y_i)$$

$$\mathcal{L}_{\text{HMM}}(\boldsymbol{\vartheta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\vartheta}(y_i | \phi_{t_i}) p_{\vartheta}(\phi_{t_i} | \phi_{t_{i-1}})$$

Calculating the CTC loss for “cat and hat”

Calculating the gradient of the CTC loss

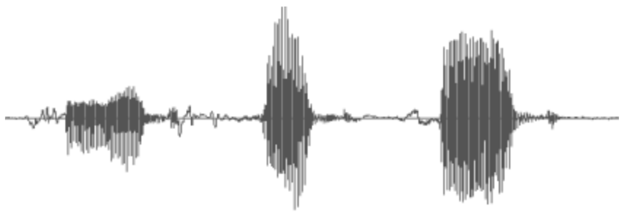


$$\mathcal{L}_{\mathrm{CTC}}(\boldsymbol{\theta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i}|y_i)$$

$$\mathcal{L}_{\mathrm{HMM}}(\boldsymbol{\vartheta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\vartheta}(y_i|\phi_{t_i})\, p_{\vartheta}(\phi_{t_i}|\phi_{t_{i-1}})$$

$$\frac{\partial \mathcal{L}_{\mathrm{CTC}}(\boldsymbol{\theta})}{\partial p_{\theta}(\boldsymbol{\phi}|y_j)} = \frac{-1}{\sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i}|y_i)} \sum_{t:\phi_{t_j}=\boldsymbol{\phi}} \frac{1}{p_{\theta}(\boldsymbol{\phi}|y_j)} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i}|y_i)$$

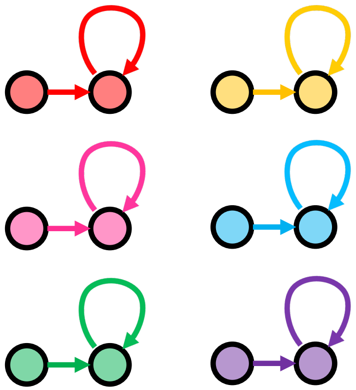
$$\frac{\partial \mathcal{L}_{\mathrm{CTC}}(\boldsymbol{\theta})}{\partial p_{\theta}(\boldsymbol{\phi}|y_j)} = -\frac{1}{p_{\theta}(\boldsymbol{\phi}|y_j)} \frac{\sum_{t:\phi_{t_j}=\boldsymbol{\phi}} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i}|y_i)}{\sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i}|y_i)}$$



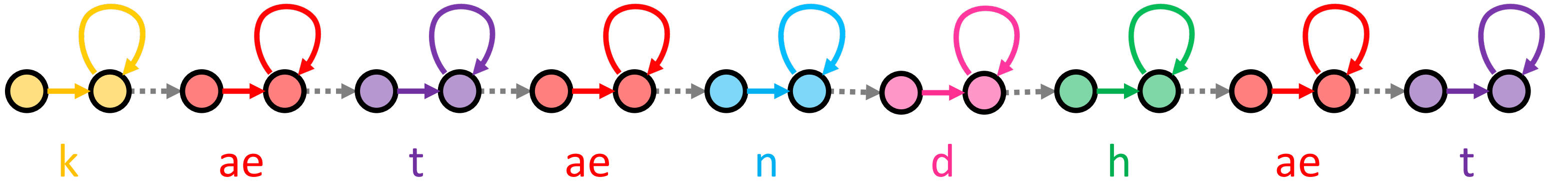
cat and hat

and	ae	n	d
cat	k	ae	t
hat	h	ae	t

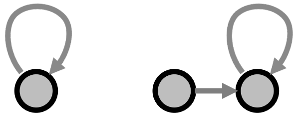
Phoneme HMMs



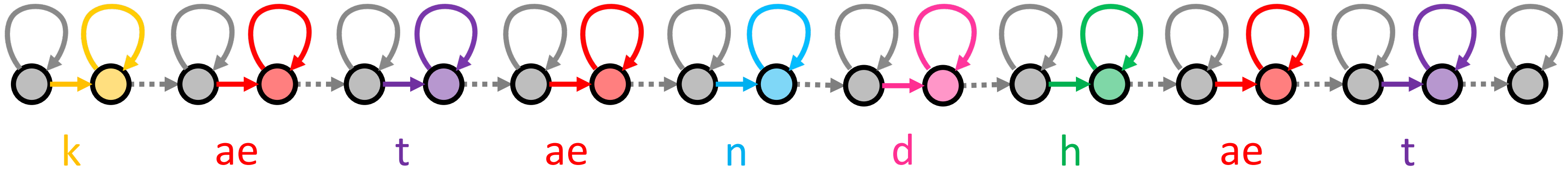
Composite HMM for “cat and hat”

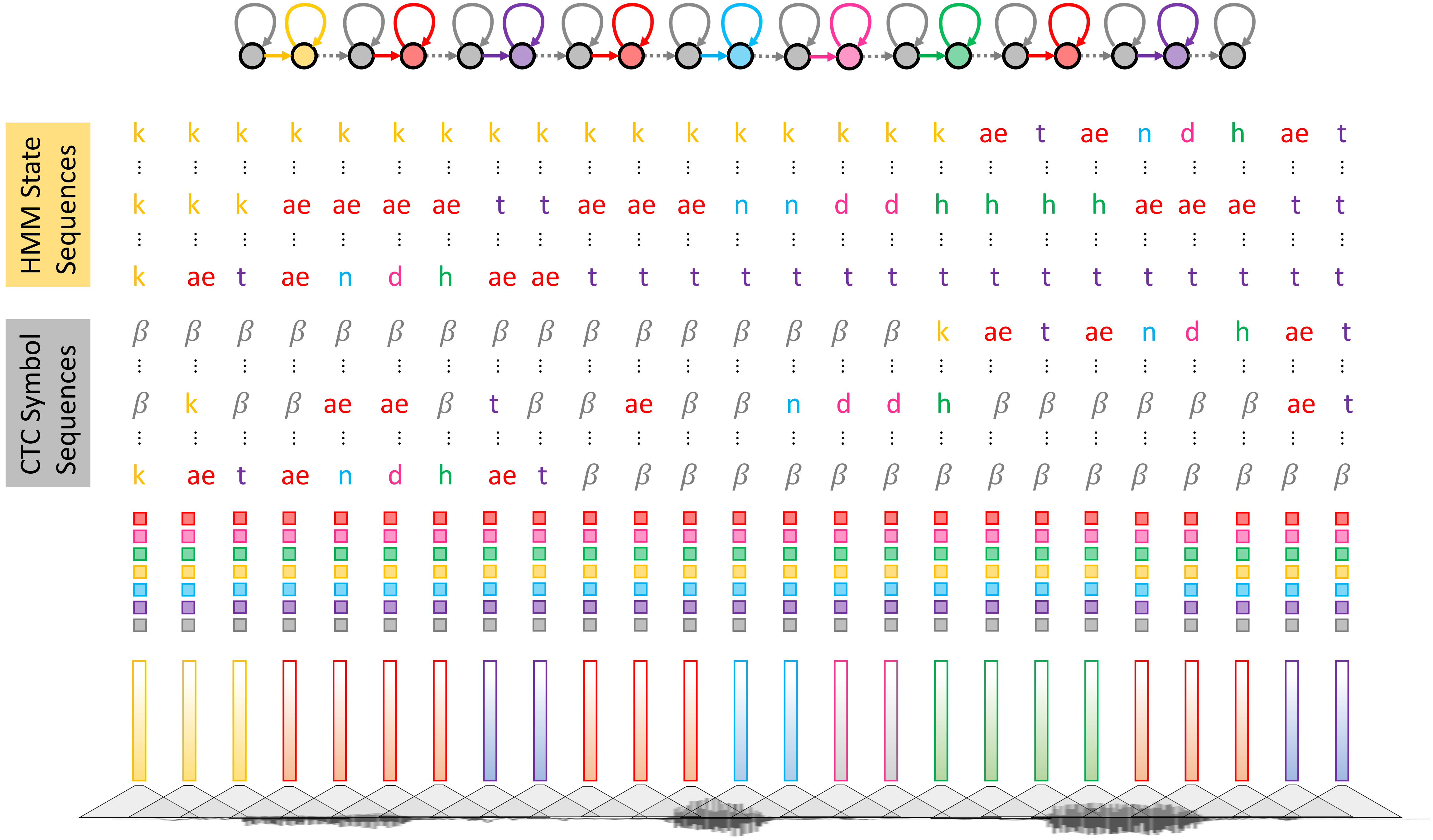


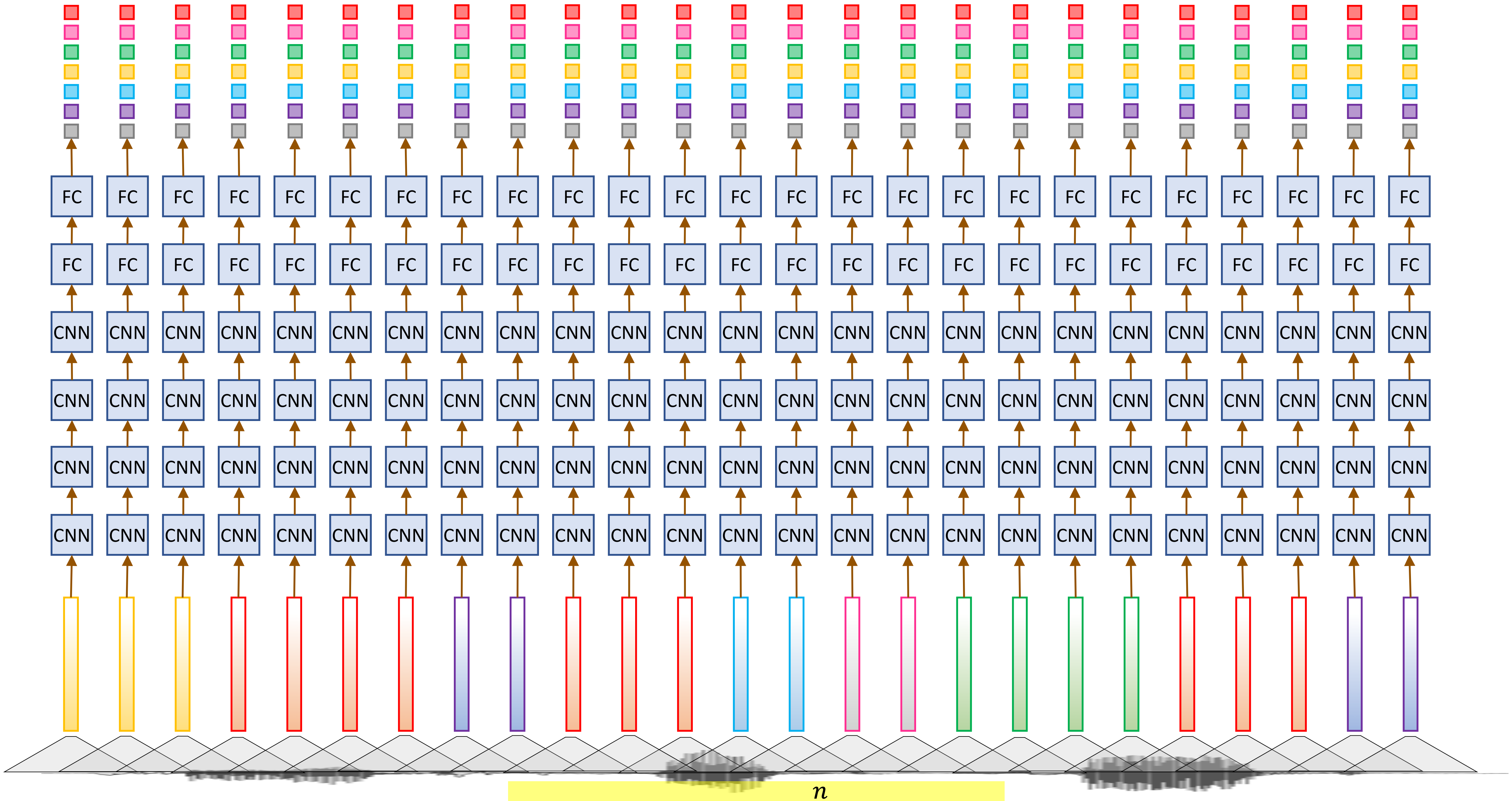
The CTC “`Blank`” Symbol (β)



FSA of permissible CTC strings for “cat and hat”




HMM State Sequences
k k k k k k k k k k k k k k k k k ae t ae n d h ae t
⋮
k k k ae ae ae ae t t ae ae ae n n d d h h h h ae ae ae t t
⋮
k ae t ae n d h ae ae t t t t t t t t t t t t t t t t
CTC Symbol Sequences
β β β β β β β β β β β β β β β β k ae t ae n d h ae t
⋮
β k β β ae ae β t β β ae β β n d d h β β β β β β ae t
⋮
k ae t ae n d h ae t β β β β β β β β β β β β β β β β

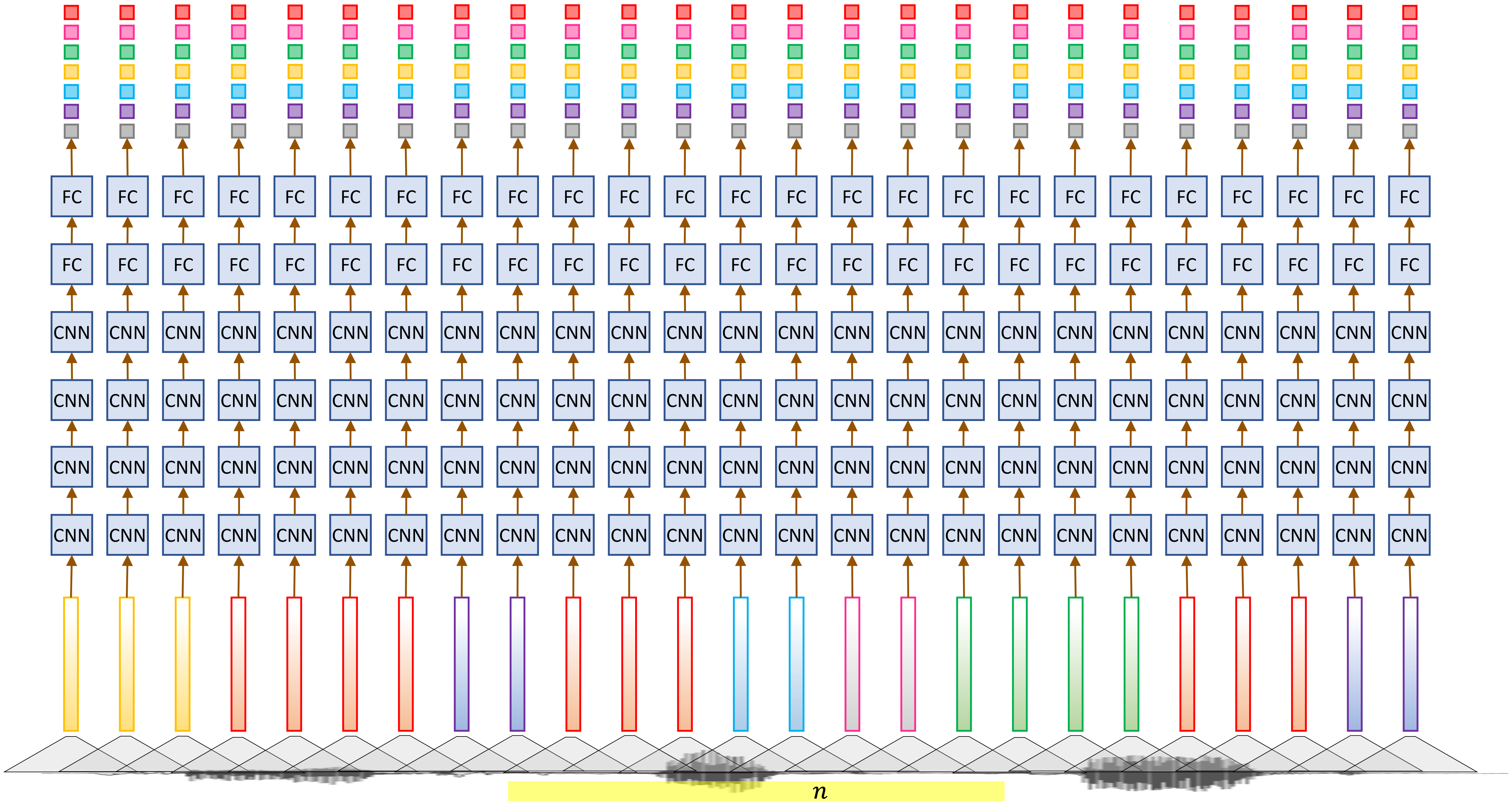


$$\mathcal{L}_{\text{CTC}}(\boldsymbol{\theta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i} | y_i)$$

End-to-End Speech Recognition using Neural Networks with Attention

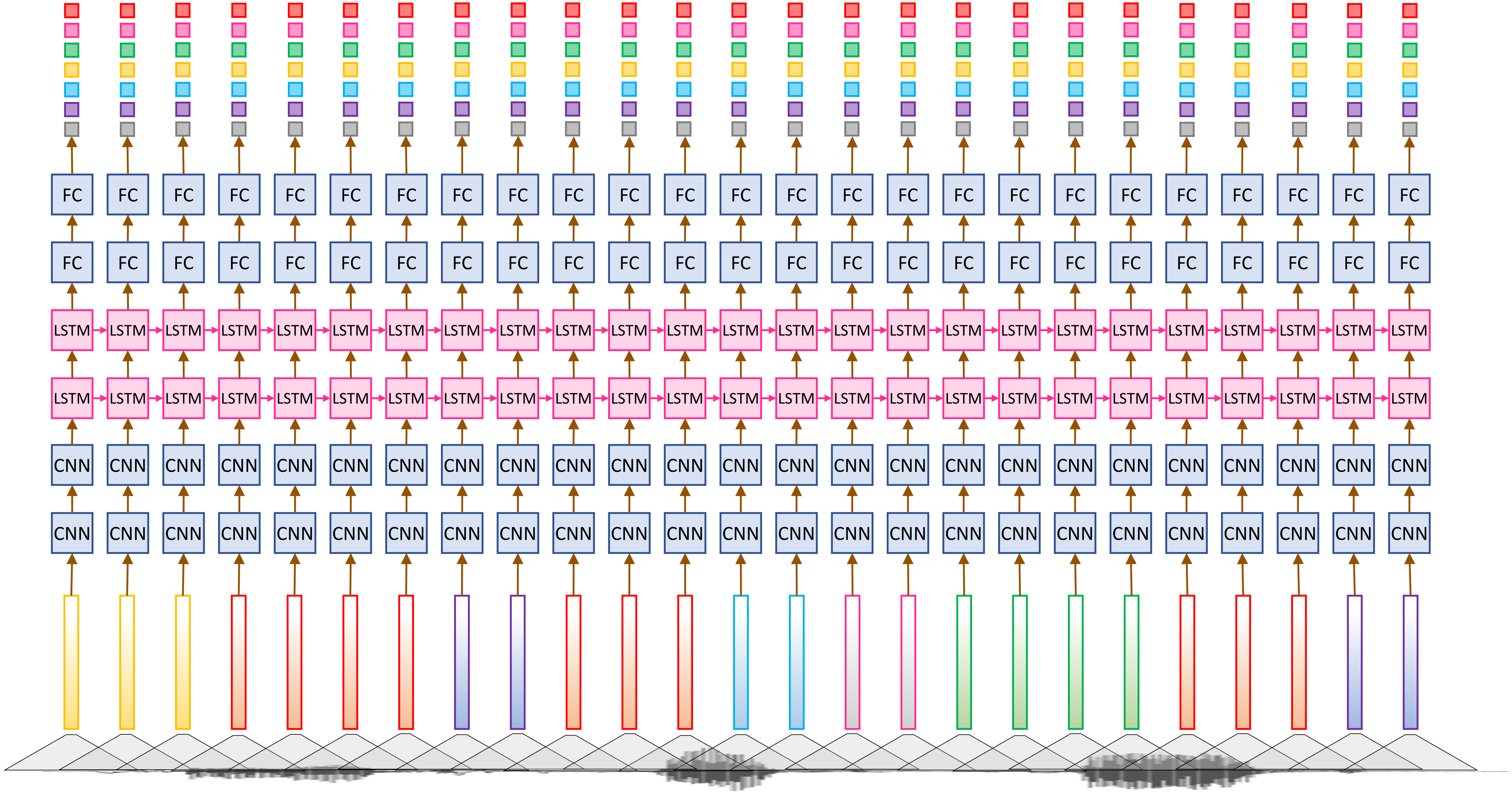
Efforts to Get Further Away from GMM-HMM Architectures

A CNN Architecture and CTC Loss

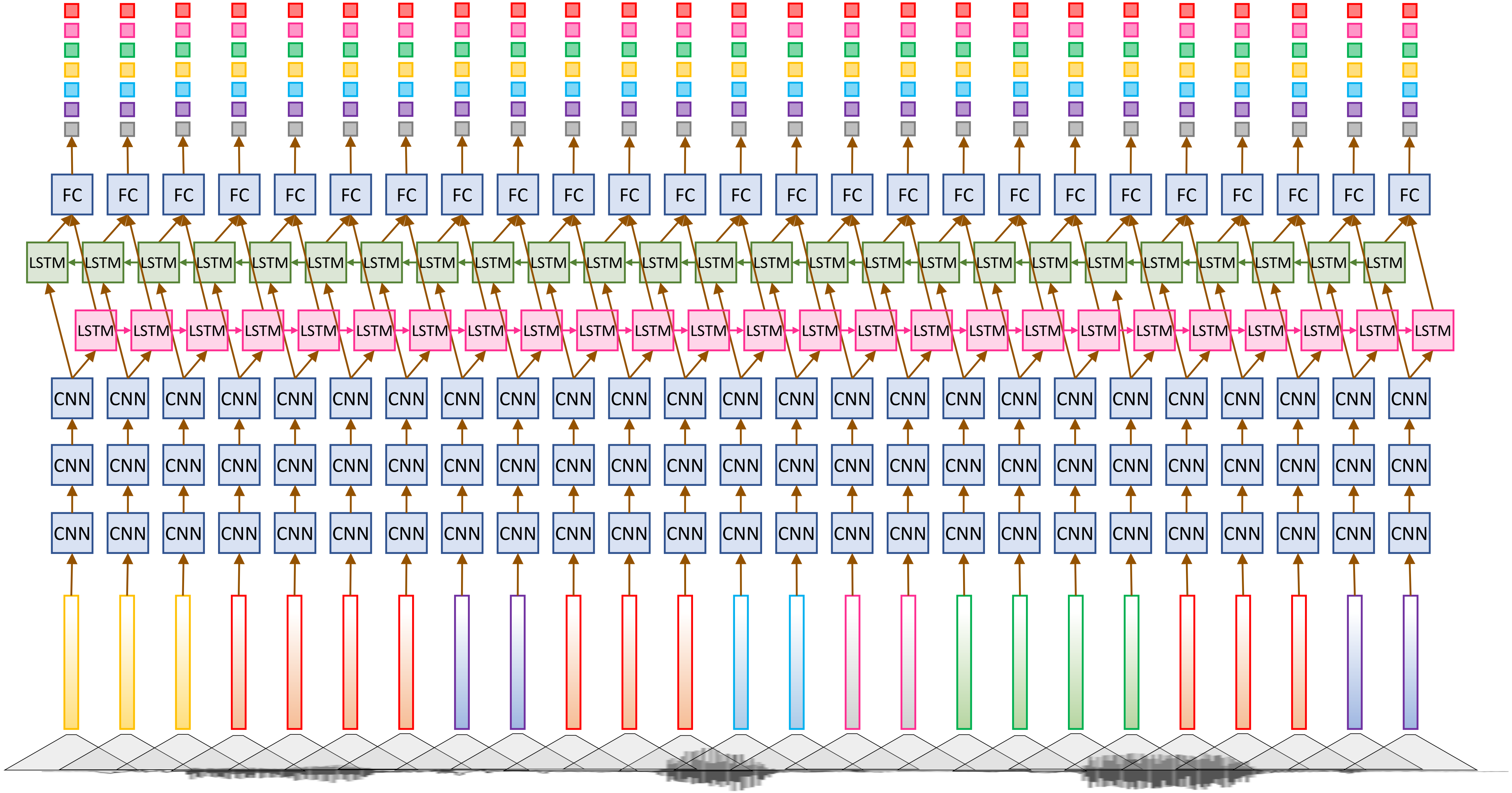


$$\mathcal{L}_{\mathrm{CTC}}(\boldsymbol{\theta}) = -\log \sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_i} | y_i)$$

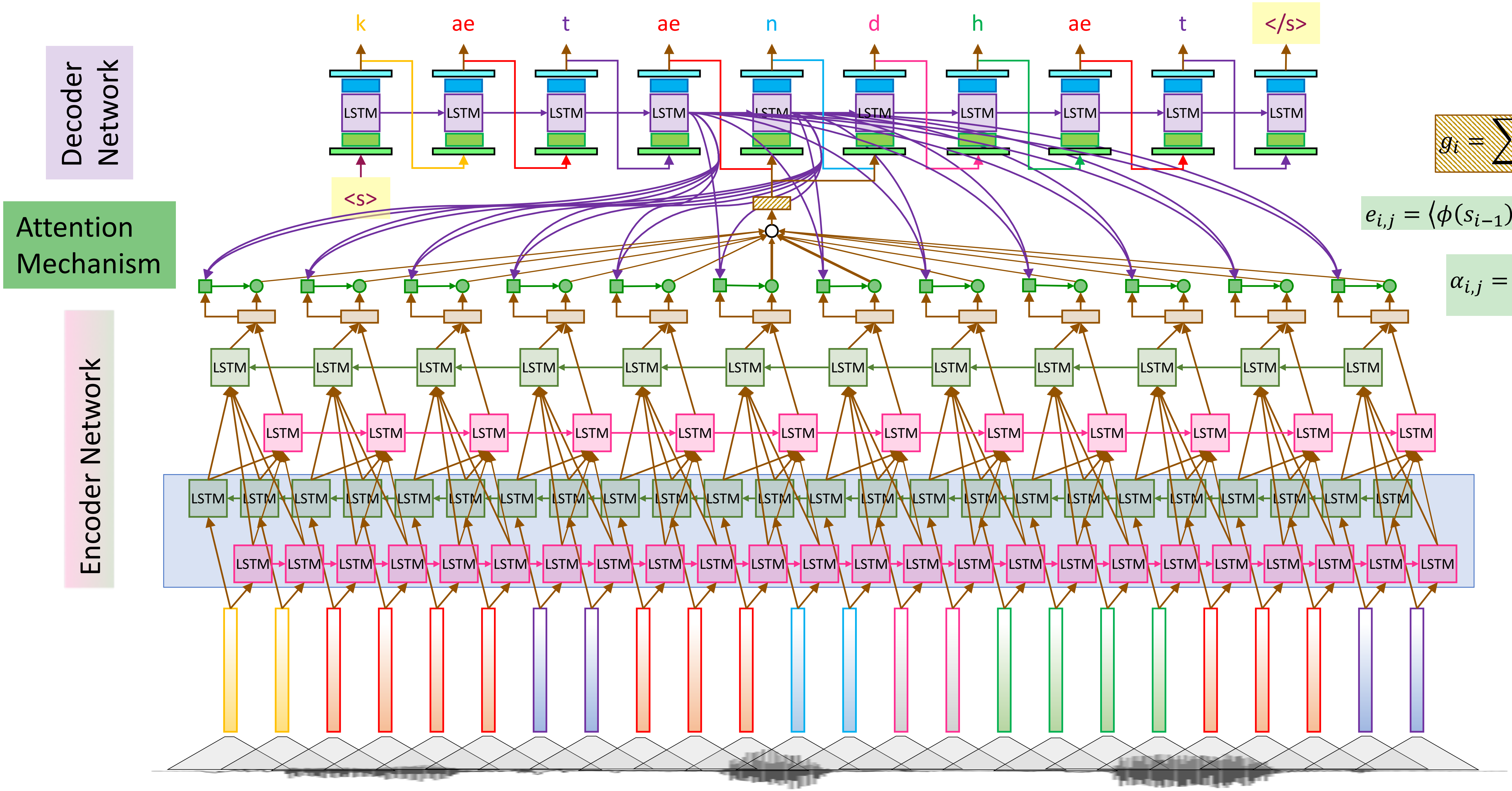
A CNN+LSTM Architecture



A Bidirectional LSTM Architecture (Deep Speech)



An Encoder-Decoder Architecture with Attention



$$g_i = \sum_j \alpha_{i,j} h_j$$

$$e_{i,j} = \langle \phi(s_{i-1}), \psi(h_j) \rangle$$

$$\alpha_{i,j} = \frac{\exp e_{i,j}}{\sum_{j'} e_{i,j'}}$$

Summary + Q&A