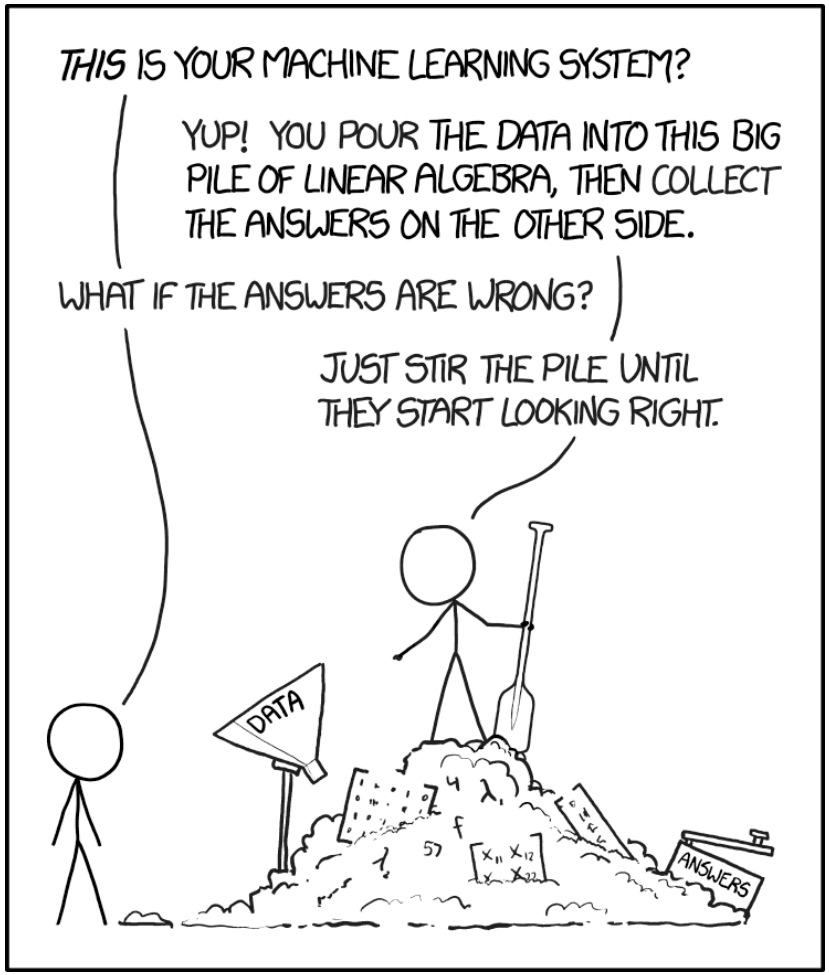
(How) should we use domain knowledge in the era of deep learning?

Karen Livescu

Toyota Technological Institute at Chicago



https://xkcd.com/1838/

Should we use domain knowledge in the era of deep learning?

Should we use domain knowledge in the era of deep learning?

An imaginary conversation...

Should we use domain knowledge in the era of deep learning?

An imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.

Should we use domain knowledge in the era of deep learning?

An imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- **Alice**: But what if we don't have enough data?

Should we use domain knowledge in the era of deep learning?

An imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- **Alice**: But what if we don't have enough data?
- **Bob**: We get more.

Should we use domain knowledge in the era of deep learning?

An imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- **Alice**: But what if we don't have enough data?
- **Bob**: We get more.
- **Alice**: But what if that's hard?

Should we use domain knowledge in the era of deep learning?

An imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- **Alice**: But what if we don't have enough data?
- **Bob**: We get more.
- **Alice**: But what if that's hard?
- **Bob**: We try harder.

Should we use domain knowledge in the era of deep learning?

Should we use domain knowledge in the era of deep learning?

The next day...

Should we use domain knowledge in the era of deep learning?

The next day...

- **Alice**: Deep networks have so many hyperparameters and are so hard to optimize. How do you decide on the number of layers, size of layers, DNN vs. CNN vs. RNN, ResNet vs. FractalNet vs. QuantumNet? SGD vs. AdaGrad vs. Adam vs. rmsprop?

Should we use domain knowledge in the era of deep learning?

The next day...

- **Alice**: Deep networks have so many hyperparameters and are so hard to optimize. How do you decide on the number of layers, size of layers, DNN vs. CNN vs. RNN, ResNet vs. FractalNet vs. QuantumNet? SGD vs. AdaGrad vs. Adam vs. rmsprop?
- **Bob**: I don't. I throw them all onto the GPUs and go read xkcd.

Should we use domain knowledge in the era of deep learning?

The next day...

- **Alice**: Deep networks have so many hyperparameters and are so hard to optimize. How do you decide on the number of layers, size of layers, DNN vs. CNN vs. RNN, ResNet vs. FractalNet vs. QuantumNet? SGD vs. AdaGrad vs. Adam vs. rmsprop?
- **Bob**: I don't. I throw them all onto the GPUs and go read xkcd.
- **Alice**: But other people need the GPUs too...

Should we use domain knowledge in the era of deep learning?

The next day...

- **Alice**: Deep networks have so many hyperparameters and are so hard to optimize. How do you decide on the number of layers, size of layers, DNN vs. CNN vs. RNN, ResNet vs. FractalNet vs. QuantumNet? SGD vs. AdaGrad vs. Adam vs. rmsprop?
- **Bob**: I don't. I throw them all onto the GPUs and go read xkcd.
- **Alice**: But other people need the GPUs too...
- **Bob**: Yoav will buy more GPUs.

Should we use domain knowledge in the era of deep learning?

The next day...

- **Alice**: Deep networks have so many hyperparameters and are so hard to optimize. How do you decide on the number of layers, size of layers, DNN vs. CNN vs. RNN, ResNet vs. FractalNet vs. QuantumNet? SGD vs. AdaGrad vs. Adam vs. rmsprop?
- **Bob**: I don't. I throw them all onto the GPUs and go read xkcd.
- **Alice**: But other people need the GPUs too...
- **Bob**: Yoav will buy more GPUs.
- **Alice**: But think about all the wasted cycles! Air conditioning! The environment!

Should we use domain knowledge in the era of deep learning?

The next day...

- **Alice**: Deep networks have so many hyperparameters and are so hard to optimize. How do you decide on the number of layers, size of layers, DNN vs. CNN vs. RNN, ResNet vs. FractalNet vs. QuantumNet? SGD vs. AdaGrad vs. Adam vs. rmsprop?
- **Bob**: I don't. I throw them all onto the GPUs and go read xkcd.
- **Alice**: But other people need the GPUs too...
- **Bob**: Yoav will buy more GPUs.
- **Alice**: But think about all the wasted cycles! Air conditioning! The environment!
- **Bob**: Environment, schmenvironment. It won't matter after the singularity.

Should we use domain knowledge in the era of deep learning?

The following day...

Should we use domain knowledge in the era of deep learning?

The following day...

- **Vision Alice**: What do I do with all this knowledge I have about geometry, light diffusion, ...?

Should we use domain knowledge in the era of deep learning?

The following day...

- **Vision Alice**: What do I do with all this knowledge I have about geometry, light diffusion, ...?
- **Speech Alice**: Phonemes, triphones, articulatory features, ...?

Should we use domain knowledge in the era of deep learning?

The following day...

- **Vision Alice**: What do I do with all this knowledge I have about geometry, light diffusion, ...?
- **Speech Alice**: Phonemes, triphones, articulatory features, ...?
- **NLP Alice**: Syntax, morphology, X-bar theory, ...?

Should we use domain knowledge in the era of deep learning?

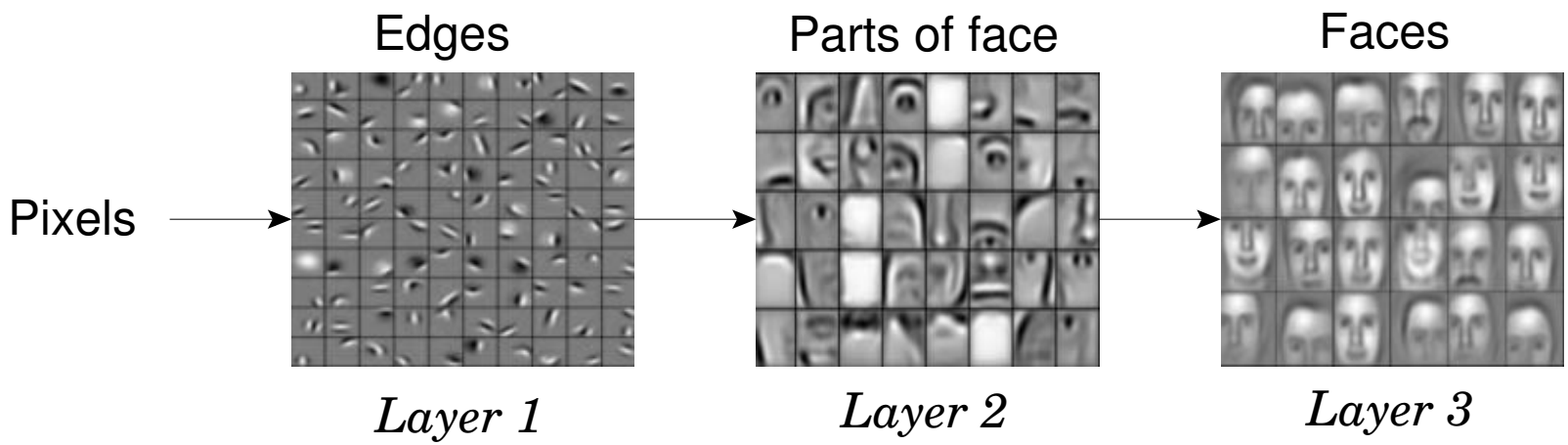
The following day...

- **Vision Alice**: What do I do with all this knowledge I have about geometry, light diffusion, ...?
- **Speech Alice**: Phonemes, triphones, articulatory features, ...?
- **NLP Alice**: Syntax, morphology, X-bar theory, ...?
- **Bob**: Forget all that. Deep networks automatically learn it. Some of that stuff is wrong anyway.

Analyzing neural representations: Vision

Analyzing neural representations: Vision

- Different layers tend to specialize for different sub-tasks (e.g., [Mohamed+ 2012, Zeiler+ 2014])
- Lower layers focus on lower-level representations
- Higher layers focus on higher-level representations



[Figure from Lee+ 2009]

Analyzing neural representations: Speech

Analyzing neural representations: Speech

In a speech recognizer [Belinkov & Glass 2017]

- Network activations are most useful for phonetic classification around the 3^{rd}-to-last layer

Analyzing neural representations: Speech

In a speech recognizer [Belinkov & Glass 2017]

- Network activations are most useful for phonetic classification around the 3^{rd}-to-last layer

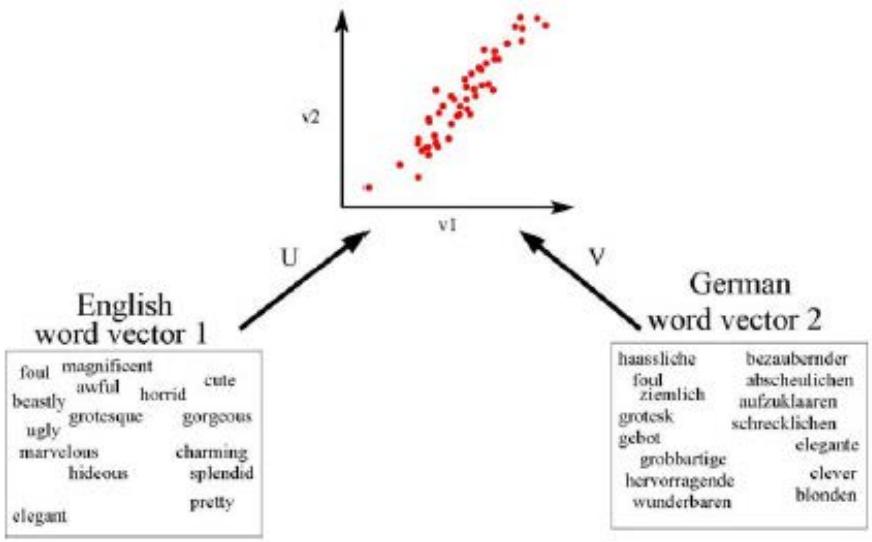
In a joint semantic model of speech and images [Chrupala+ 2017]

- The higher the layer, the better we get at homonym disambiguation (e.g., "weather" vs. "whether") and semantic similarity
- But phonetic classification accuracy peaks at an intermediate layer

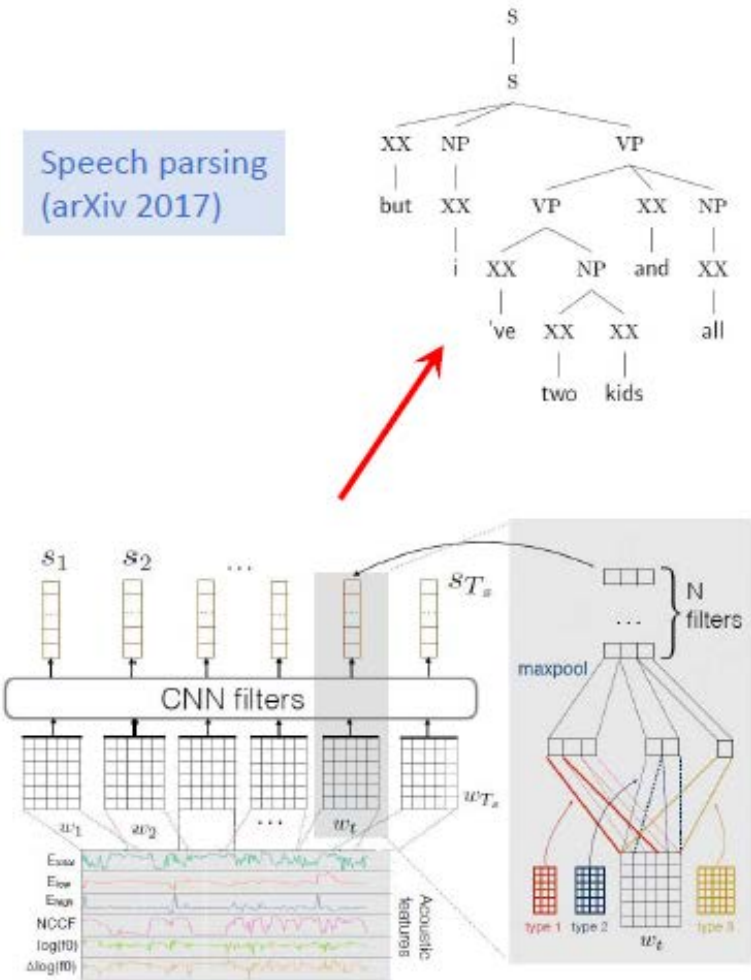
Fun with cross-domain research

Fun with cross-domain research

Multi-lingual word embeddings (NAACL 2015)



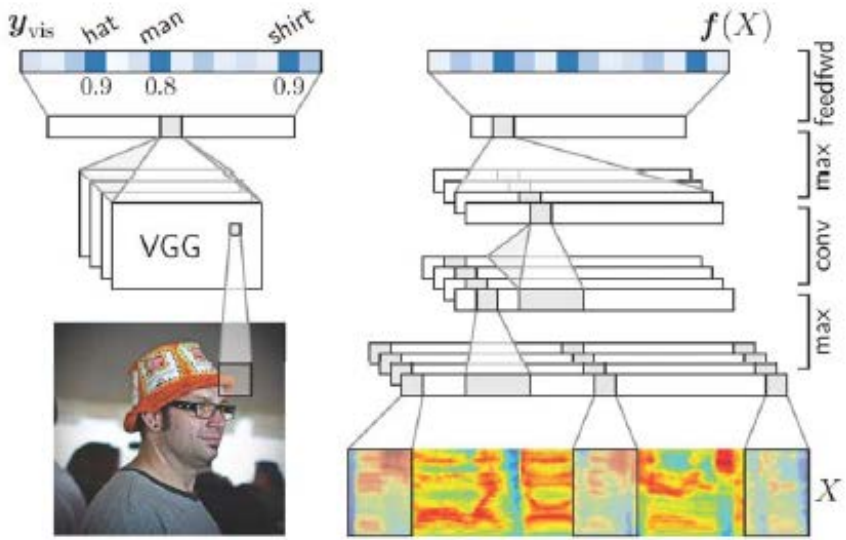
Speech parsing (arXiv 2017)



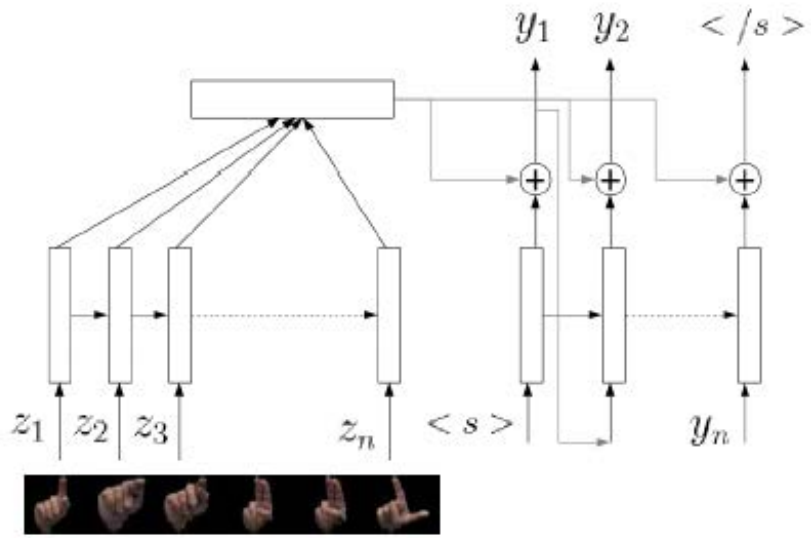
Fun with cross-domain research

Fun with cross-domain research

Learning spoken concepts from speech + images (Interspeech 2017)



Automatic sign language recognition (*CSL* 2017, ASRU 2017)



This talk: Focus on speech tasks

This talk: Focus on speech tasks

Input: Speech waveform

This talk: Focus on speech tasks

Input: Speech waveform

Output:

- **What was said? (automatic speech recognition)** (this talk)
- Does it contain the term "New Jersey"? (keyword spotting)
- What is it about? (topic ID)
- Is the speaker angry? (emotion ID)
- Was the pronunciation correct? (proficiency testing)
- How do you say it in German? (speech translation)
- ...

Back to the imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- ...

Back to the imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- ...

There are many settings where data is likely to remain scarce

Back to the imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- ...

There are many settings where data is likely to remain scarce

- Most of the world's $\sim$7000 languages are poorly (or not) documented

Back to the imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- ...

There are many settings where data is likely to remain scarce

- Most of the world's $\sim$7000 languages are poorly (or not) documented
- Many atypical speech styles: accents, speech disorders, inebriation, ...

Back to the imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- ...

There are many settings where data is likely to remain scarce

- Most of the world's $\sim$7000 languages are poorly (or not) documented
- Many atypical speech styles: accents, speech disorders, inebriation, ...
- Some kinds of data are unethical or illegal to collect

Back to the imaginary conversation...

- **Bob**: I love deep networks. They can learn anything given enough labeled data.
- ...

There are many settings where data is likely to remain scarce

- Most of the world's $\sim$7000 languages are poorly (or not) documented
- Many atypical speech styles: accents, speech disorders, inebriation, ...
- Some kinds of data are unethical or illegal to collect
- Some kinds of data are easy to collect, but not to label

Back to the imaginary conversation...

Back to the imaginary conversation...

- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.

Back to the imaginary conversation...

- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.
- **Alice**: But how do we get it into the deep networks? They are opaque!

Back to the imaginary conversation...

- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.
- **Alice**: But how do we get it into the deep networks? They are opaque!
- **Bob**: And we don't want to break their ability to learn their own internal representations.

Back to the imaginary conversation...

- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.
- **Alice**: But how do we get it into the deep networks? They are opaque!
- **Bob**: And we don't want to break their ability to learn their own internal representations.
- **Alice**: On this we agree!

Back to the imaginary conversation...

- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.
- **Alice**: But how do we get it into the deep networks? They are opaque!
- **Bob**: And we don't want to break their ability to learn their own internal representations.
- **Alice**: On this we agree!

This talk: 2 ideas for using domain knowledge in neural speech recognition

Back to the imaginary conversation...

- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.
- **Alice**: But how do we get it into the deep networks? They are opaque!
- **Bob**: And we don't want to break their ability to learn their own internal representations.
- **Alice**: On this we agree!

This talk: 2 ideas for using domain knowledge in neural speech recognition

- A hierarchical multitask learning approach (Toshniwal et al.)

Back to the imaginary conversation...

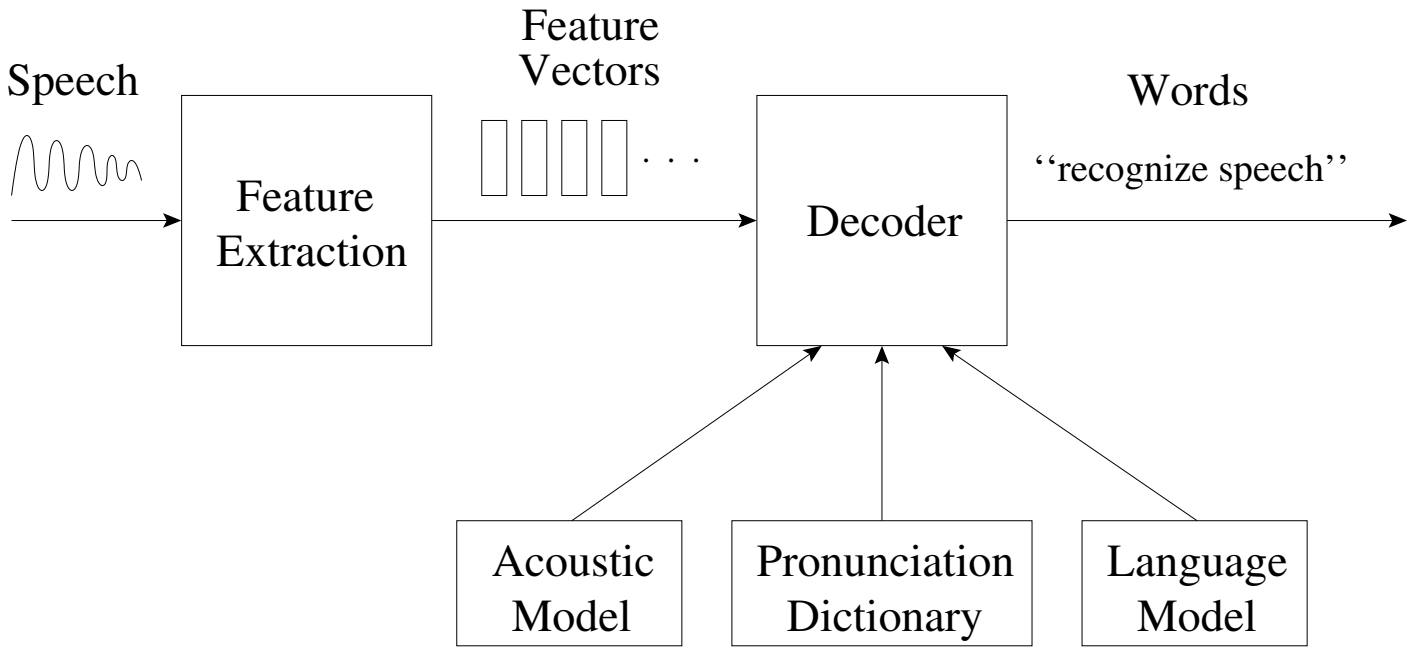
- **Bob**: OK, sometimes we won't have enough data. And then it might help to have some domain knowledge.
- **Alice**: But how do we get it into the deep networks? They are opaque!
- **Bob**: And we don't want to break their ability to learn their own internal representations.
- **Alice**: On this we agree!

This talk: 2 ideas for using domain knowledge in neural speech recognition

- A hierarchical multitask learning approach (Toshniwal et al.)
- A multi-view representation learning approach (Tang, Wang, et al.)

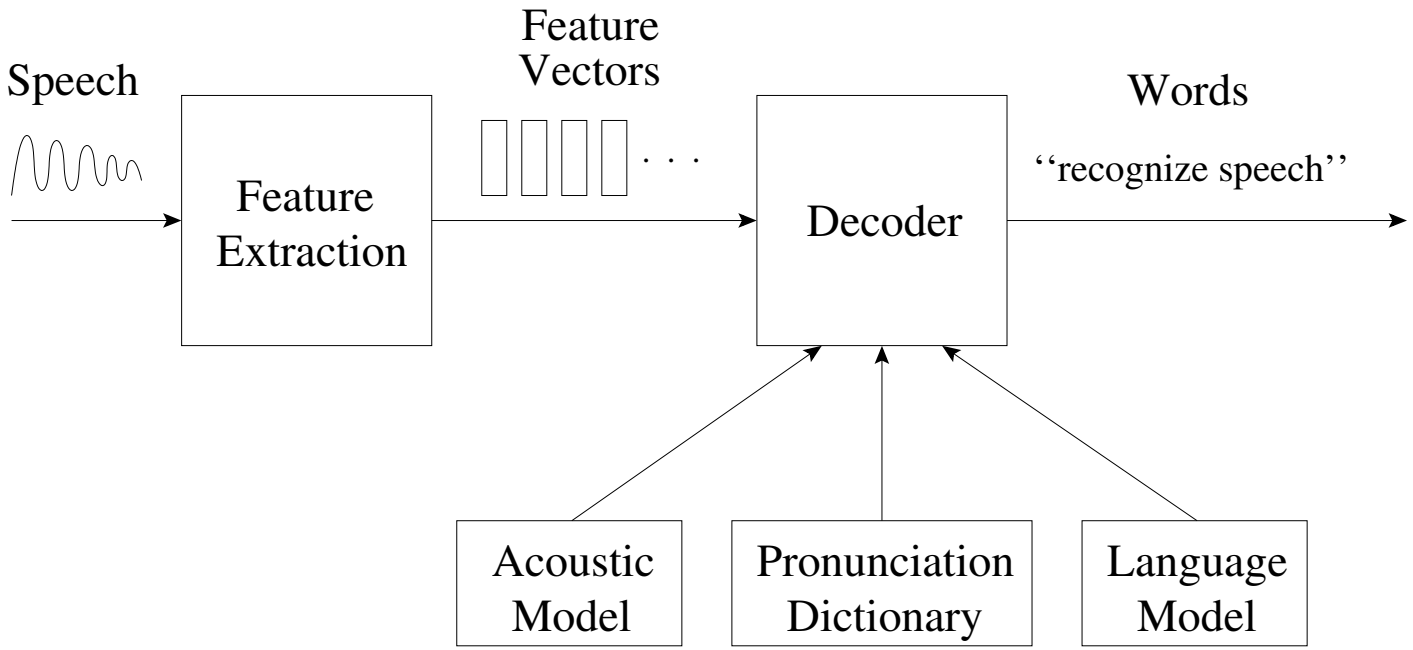
In the old days...

In the old days...



[Figure courtesy Shubham Toshniwal]

In the old days...

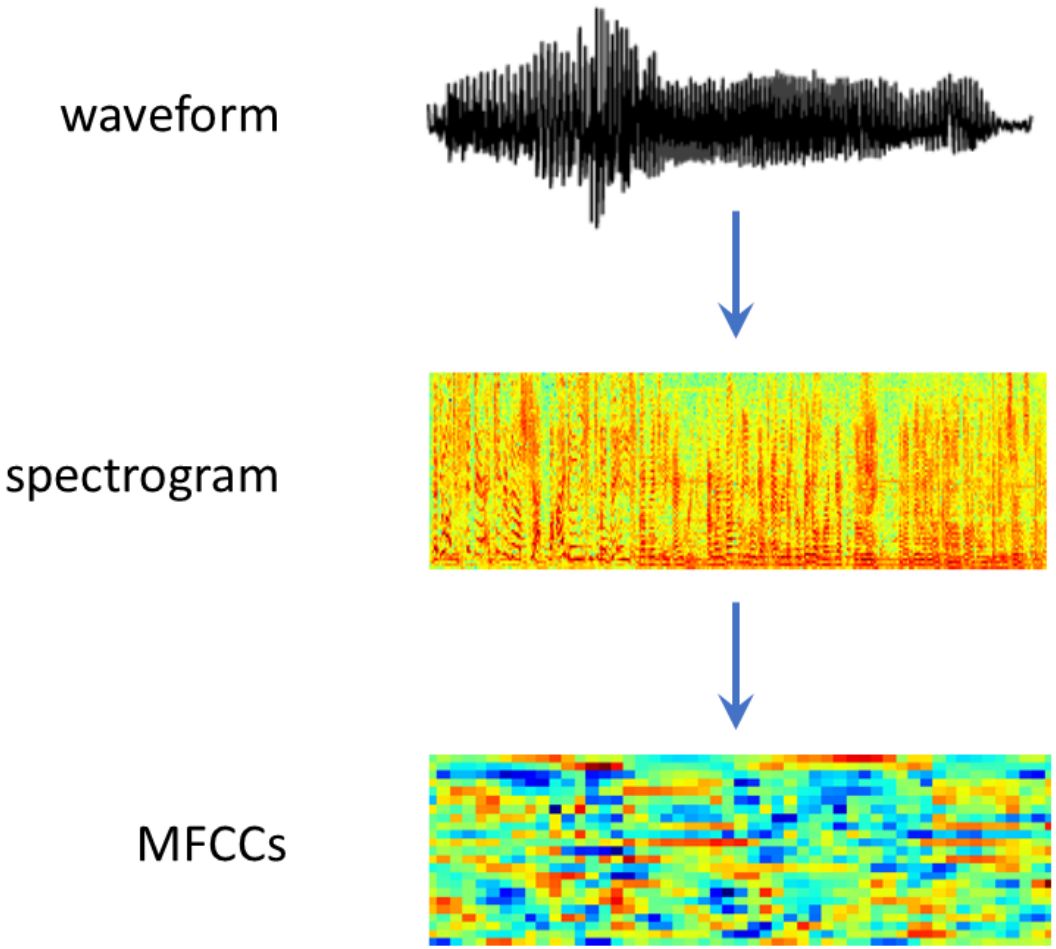


[Figure courtesy Shubham Toshniwal]

- Traditional automatic speech recognition (ASR) systems are modular
- Components correspond to different levels of representation: words, phonemes, sub-phonetic states, ...
- Components are trained separately

Acoustic features

Traditionally, inspired by properties of the human auditory system



Sub-word units

Consider the word “batrachophagous”. How do we represent its pronunciation?

Sub-word units

Consider the word “batrachophagous”. How do we represent its pronunciation?

- **Phones/phonemes:**
  [b ae t r ax k aa f ax g ax s]

Sub-word units

Consider the word “batrachophagous”. How do we represent its pronunciation?

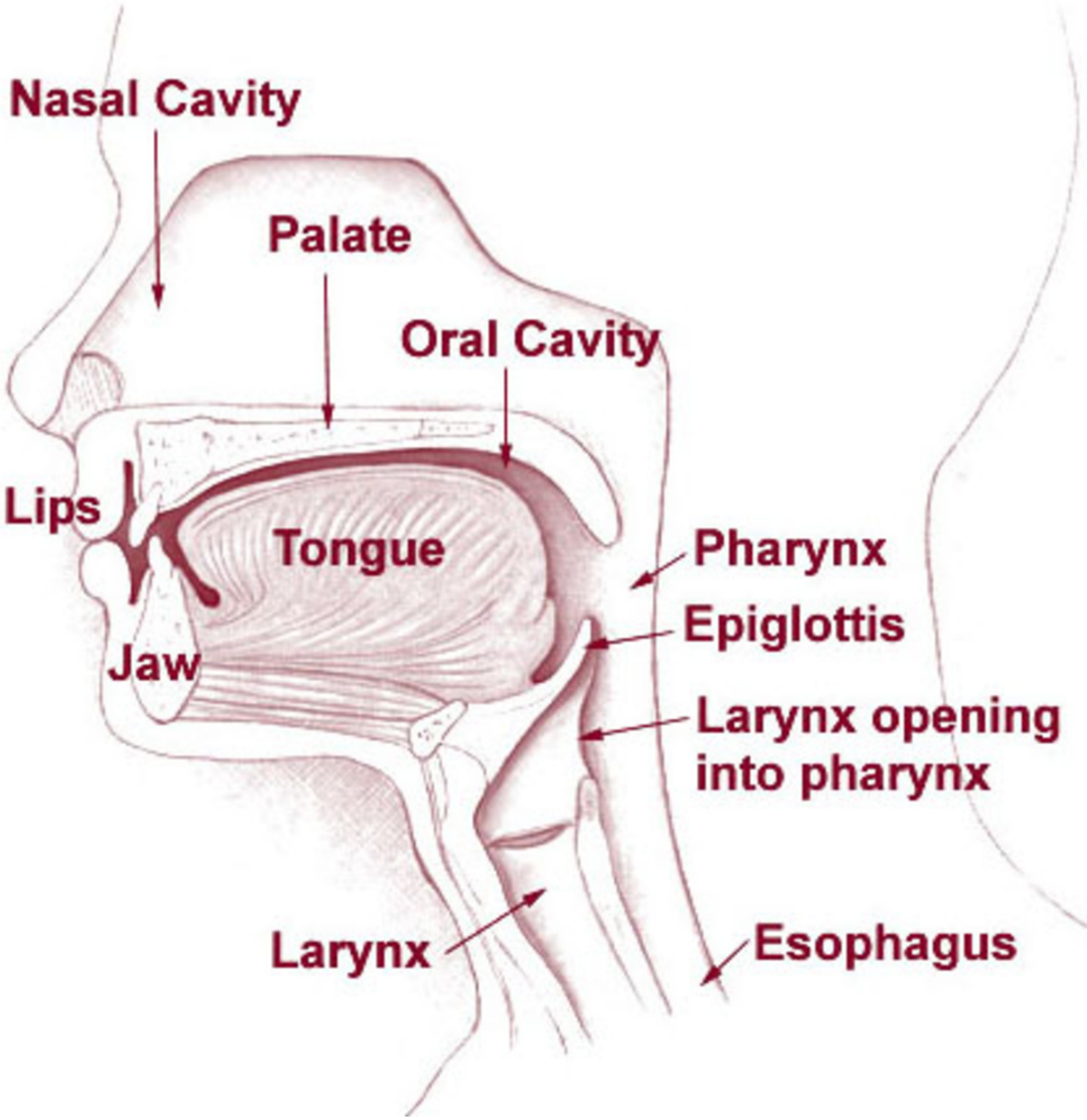
- **Phones/phonemes:**
  [b ae t r ax k aa f ax g ax s]
- **Triphones (most common):**
  [<s>+b-ae b+ae-t ae+t-r t+r-ax r+ax-k ax+k-aa k+aa-f aa+f-ax f+ax-g ax+g-ax g+ax-s ax+s-<e>]

Sub-word units

Consider the word "batrachophagous". How do we represent its pronunciation?

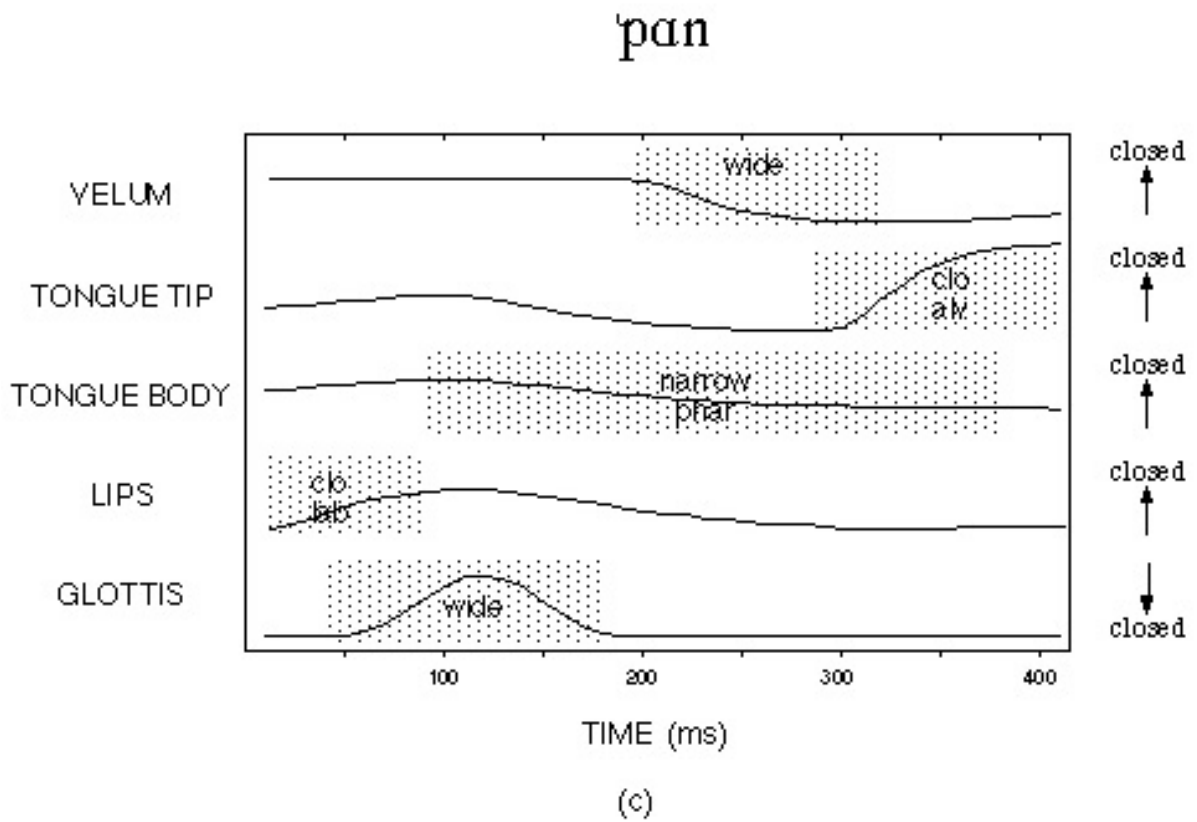
- **Phones/phonemes:**
  [b ae t r ax k aa f ax g ax s]
- **Triphones (most common):**
  [<s>+b-ae b+ae-t ae+t-r t+r-ax r+ax-k ax+k-aa k+aa-f aa+f-ax f+ax-g ax+g-ax g+ax-s ax+s-<e>]
- **Syllables:**
  [b-ae-t r-ax k-aa f-ax g-ax-s]

Sub-word models based on speech production



Sub-word models based on speech production

Articulatory phonology considers articulatory gestures to be the sub-word units

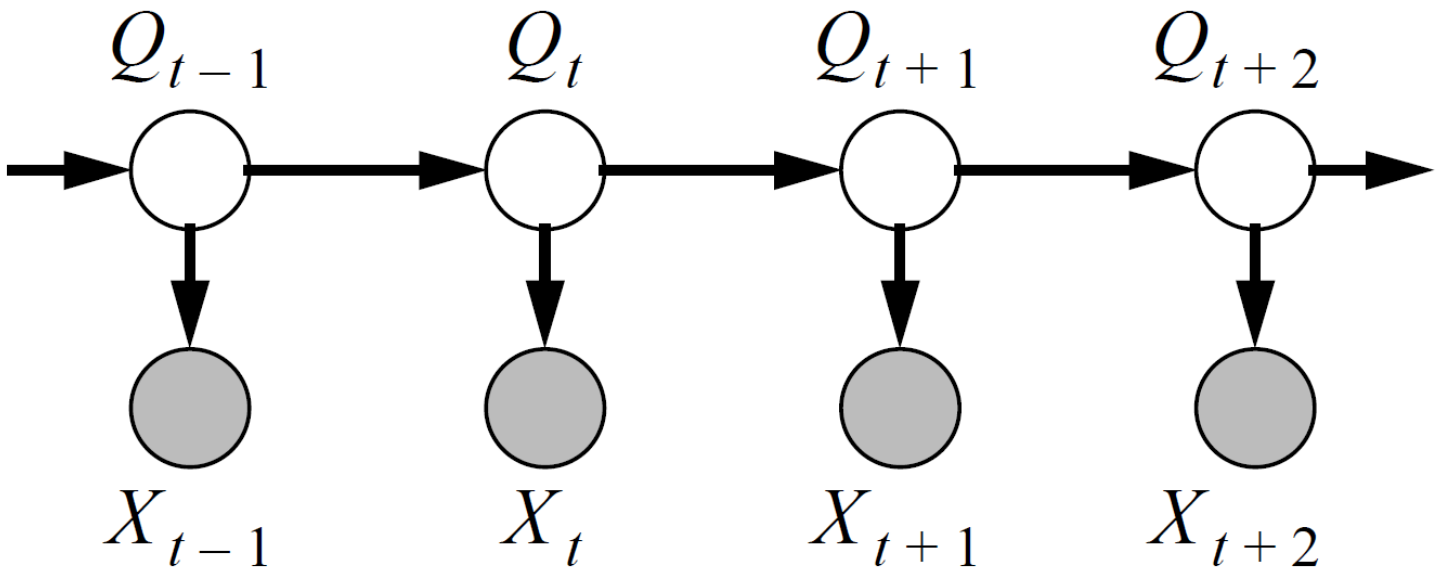


(c)

Acoustic models

Acoustic models

Hidden Markov models (HMMs) for acoustic modeling; typically one HMM per triphone



Language models

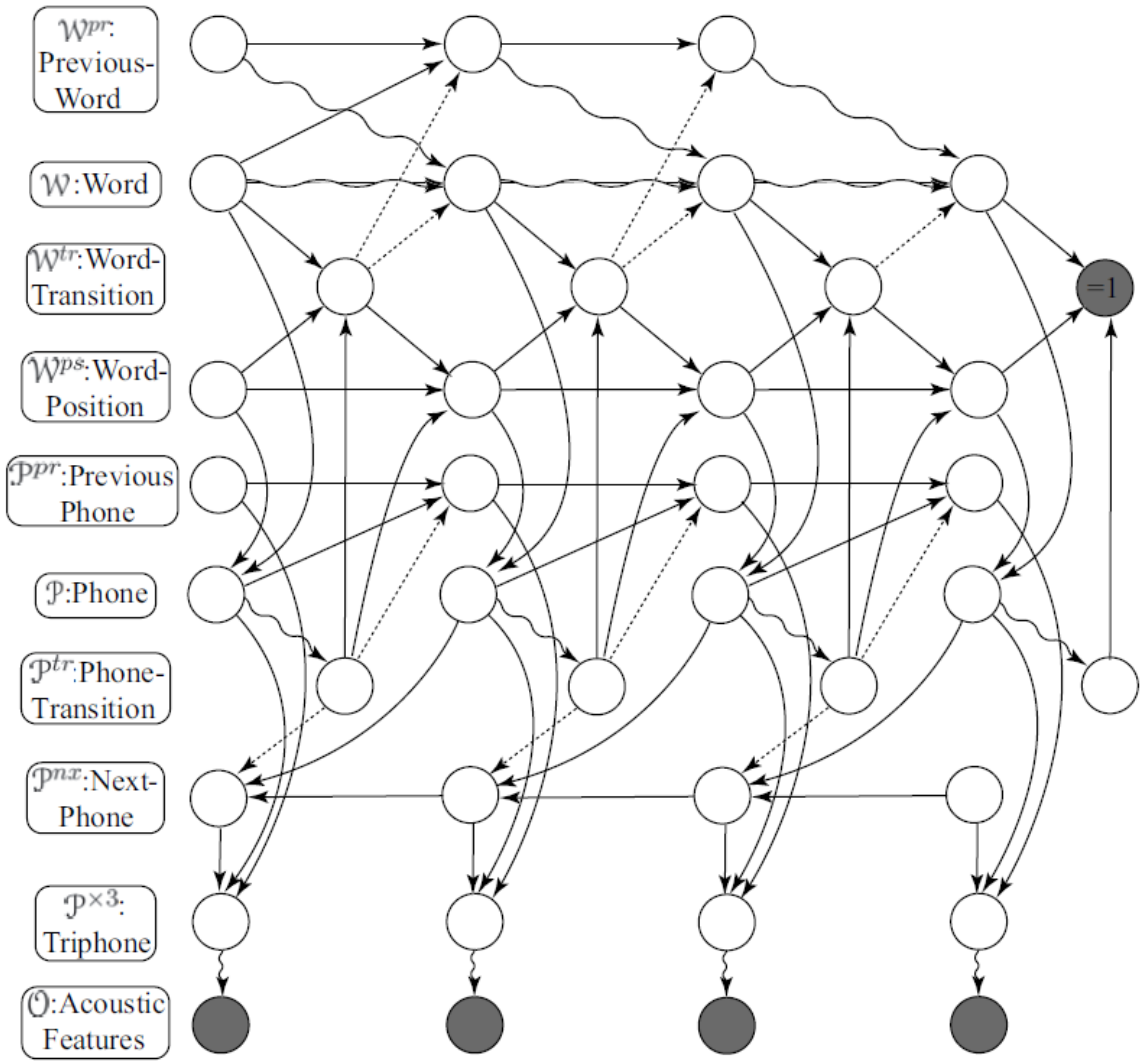
Language models

Typically some kind of Markov (n-gram) model:

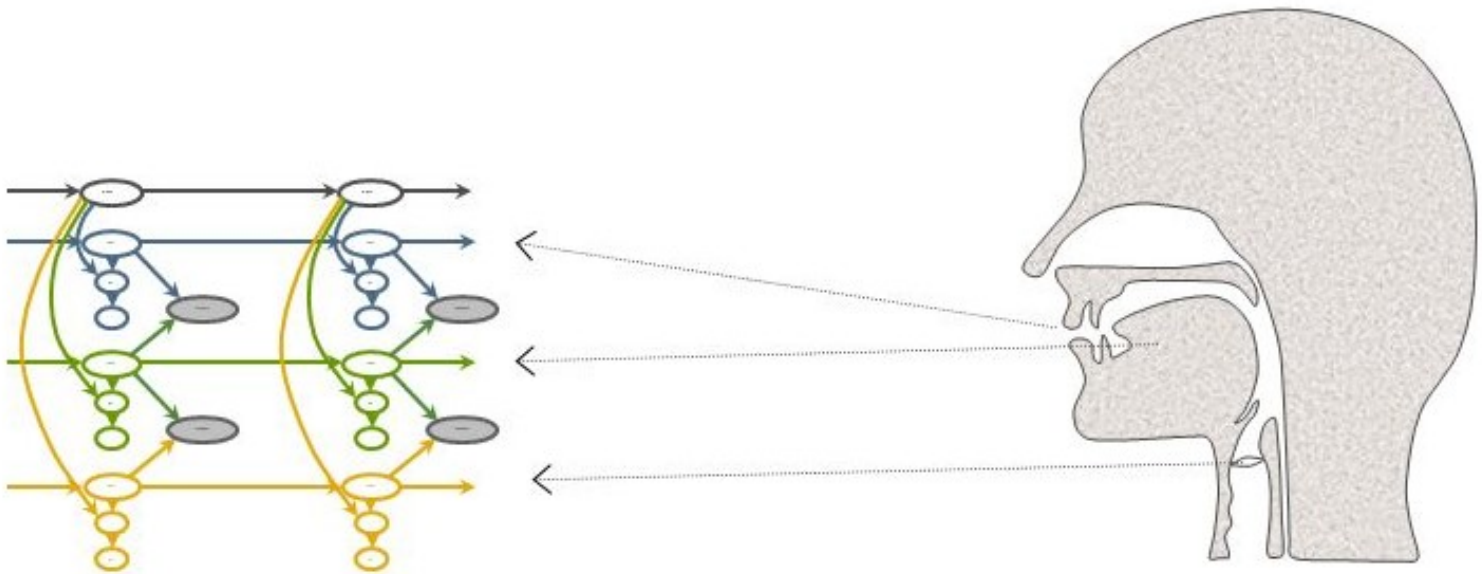
$$
\begin{aligned}
p(w_1 \dots w_L) &= \prod_{i=1}^{L} p(w_i | w_1 \dots w_{i-1}) \\
&= \prod_{i=1}^{L} p\left(w_i | w_{i-1} \dots w_{i-(n-1)}\right)
\end{aligned}
$$

A complete HMM-based recognizer [Bilmes & Bartels 2007]

Represented as a graphical model

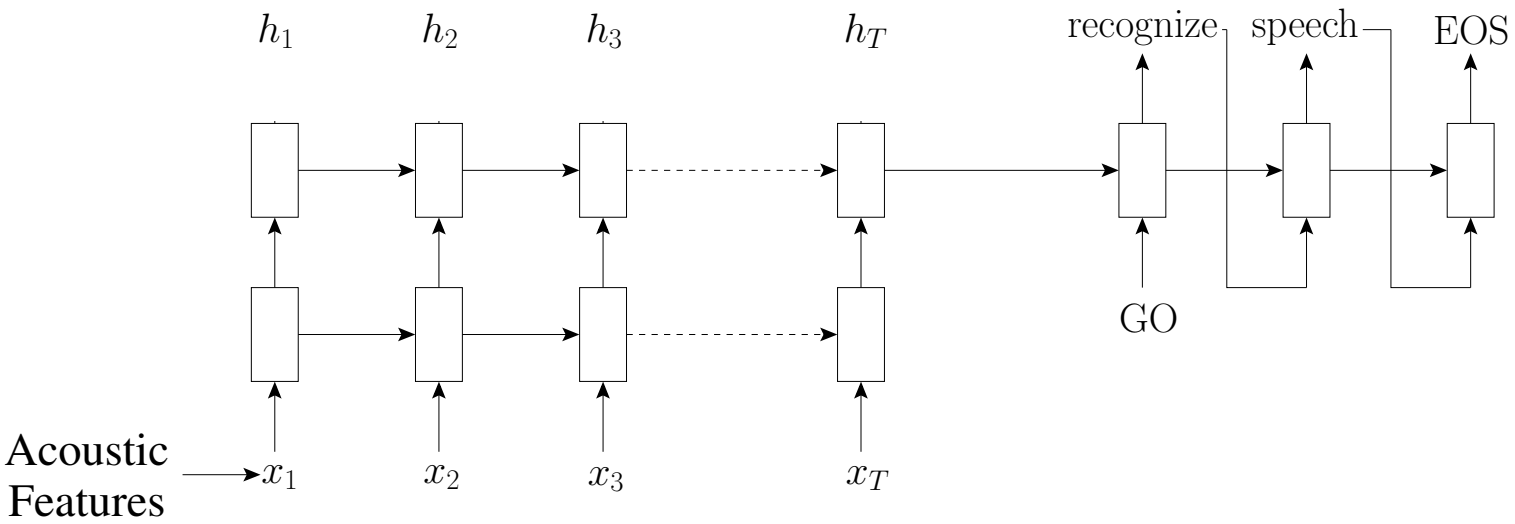


An articulatory feature-based model [Livescu 2005]

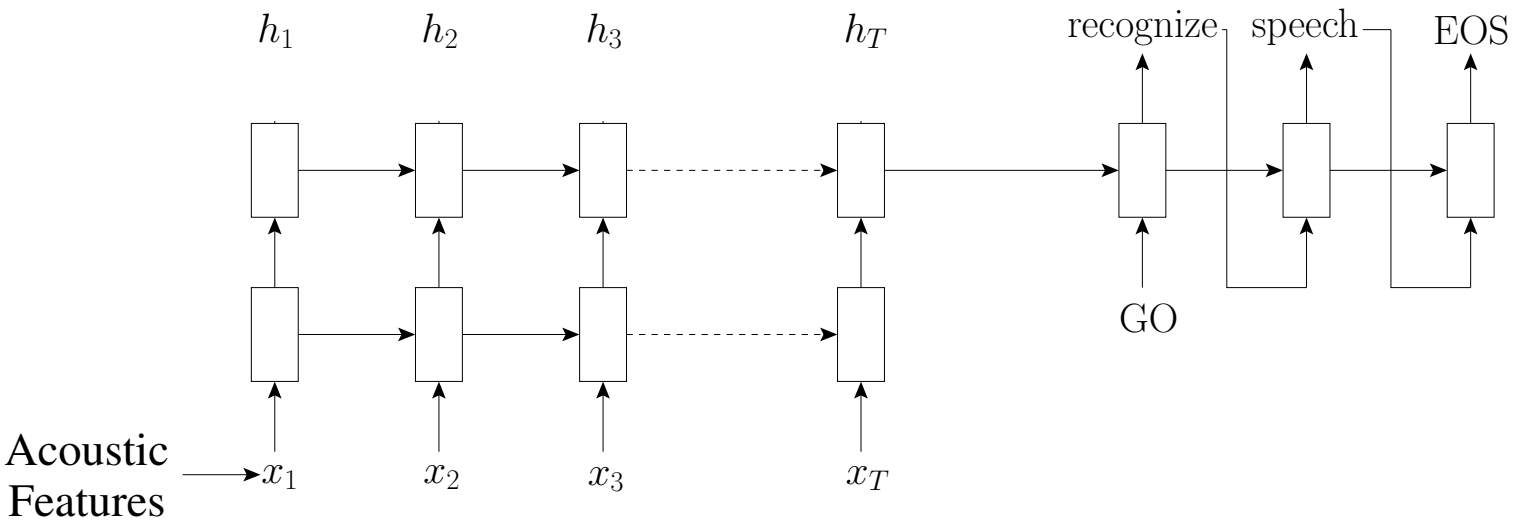


Fast forward: End-to-end neural speech recognition

Fast forward: End-to-end neural speech recognition

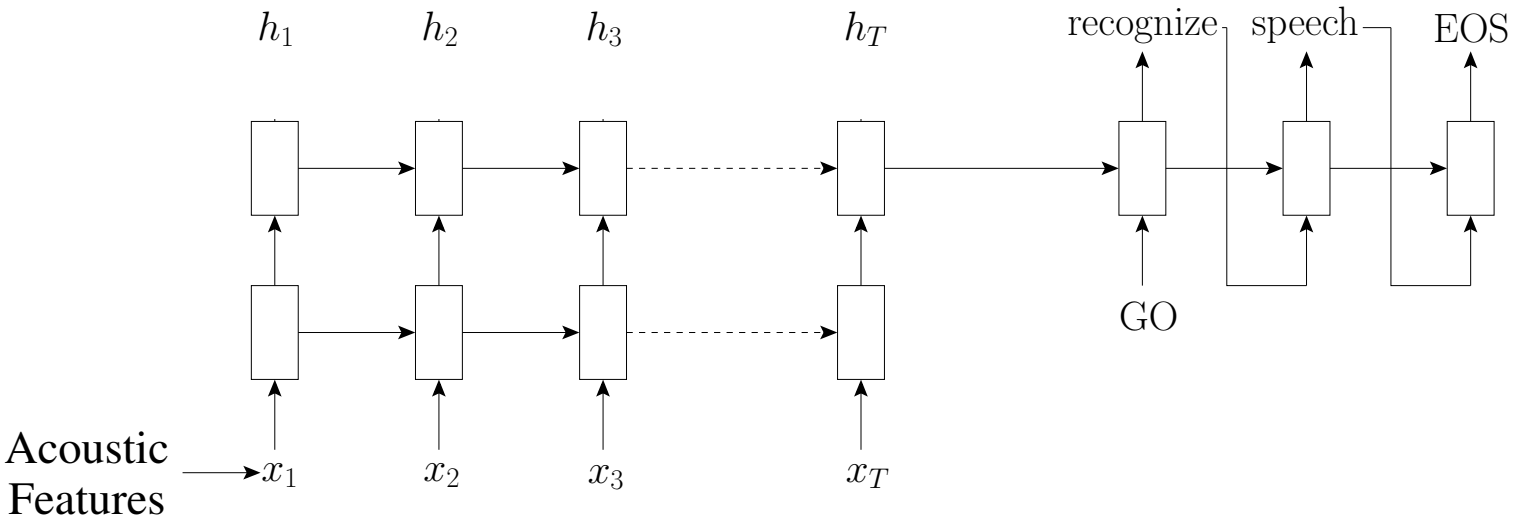


Fast forward: End-to-end neural speech recognition



All parameters learned to optimize the same task-related loss (e.g., word or character log loss)

Fast forward: End-to-end neural speech recognition



All parameters learned to optimize the same task-related loss (e.g., word or character log loss)

- No need to fix inventory of intermediate units
- Impressive results (e.g., [Zweig+ 2016])
- Similar models for other tasks: Translation [Bahdanau+ 2014], parsing [Vinyals+ 2015], image captioning [Xu+ 2015], ...

But...

End-to-end models also have some drawbacks:

- Need a lot of data
- Optimization can be challenging
- Hard to use domain knowledge about intermediate representations
- Hard to interpret intermediate representations $\Longrightarrow$ hard to debug

Idea 1: A hierarchical multitask approach

Joint work with Shubham Toshniwal, Hao Tang, Liang Lu [Interspeech 2017]

Idea 1: A hierarchical multitask approach

Joint work with Shubham Toshniwal, Hao Tang, Liang Lu [Interspeech 2017]

- Can we learn better models by explicitly encouraging meaningful intermediate representations?

Idea 1: A hierarchical multitask approach

Joint work with Shubham Toshniwal, Hao Tang, Liang Lu [Interspeech 2017]

- Can we learn better models by explicitly encouraging meaningful intermediate representations?
- Idea: Combine final task-related loss with lower-level task losses applied at lower layers

Idea 1: A hierarchical multitask approach

Joint work with Shubham Toshniwal, Hao Tang, Liang Lu [Interspeech 2017]

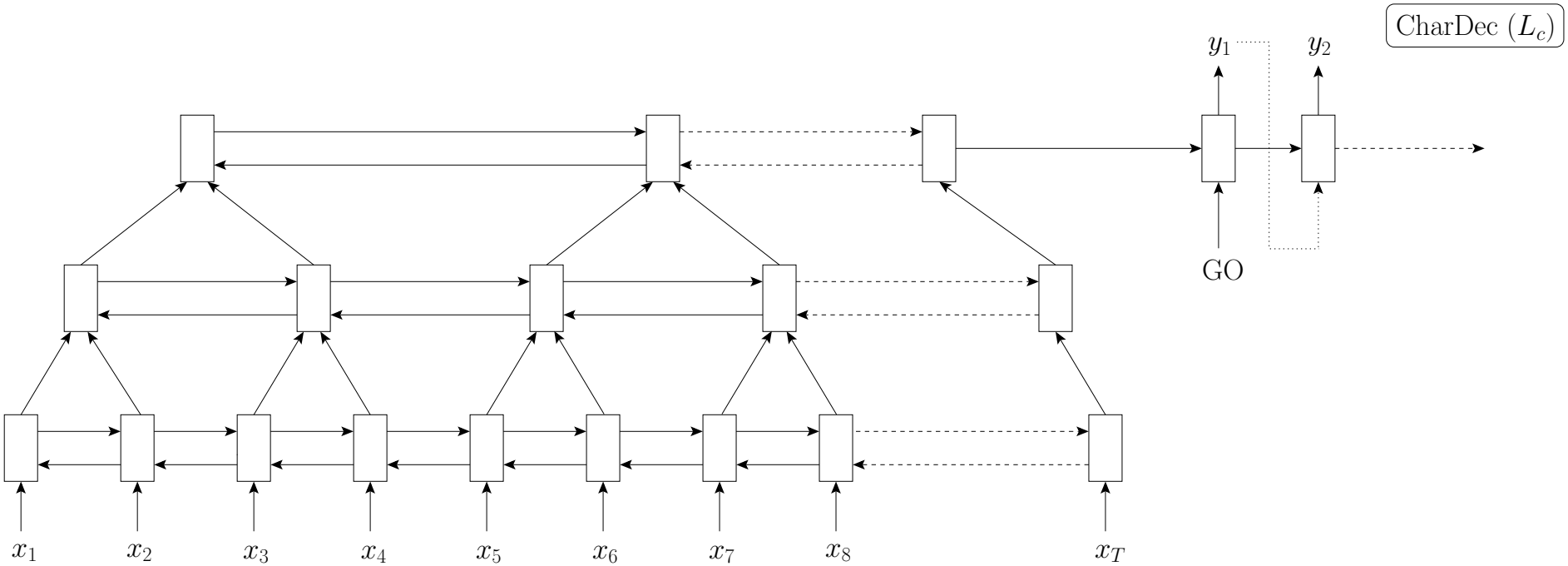
- Can we learn better models by explicitly encouraging meaningful intermediate representations?
- Idea: Combine final task-related loss with lower-level task losses applied at lower layers
  - Final task-related loss: word/character recognition
  - Lower-level tasks: phonetic recognition, HMM state classification, ...

Idea 1: A hierarchical multitask approach

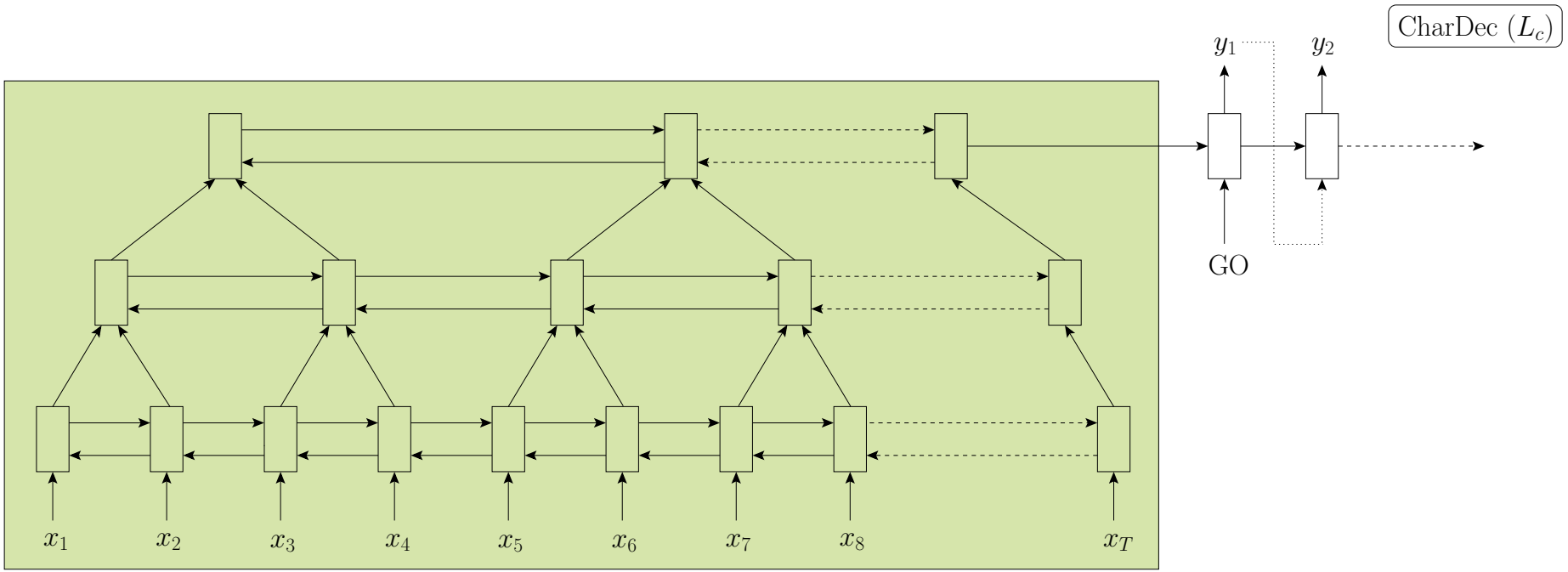
Joint work with Shubham Toshniwal, Hao Tang, Liang Lu [Interspeech 2017]

- Can we learn better models by explicitly encouraging meaningful intermediate representations?
- Idea: Combine final task-related loss with lower-level task losses applied at lower layers
  - Final task-related loss: word/character recognition
  - Lower-level tasks: phonetic recognition, HMM state classification, ...
- Similar approach recently used by others for a variety of tasks
  [Søgaard+ 2016, Hashimoto+ 2016, Weiss+ 2017, Rao+ 2017]

Baseline encoder-decoder attention model [Chan+ 2015]

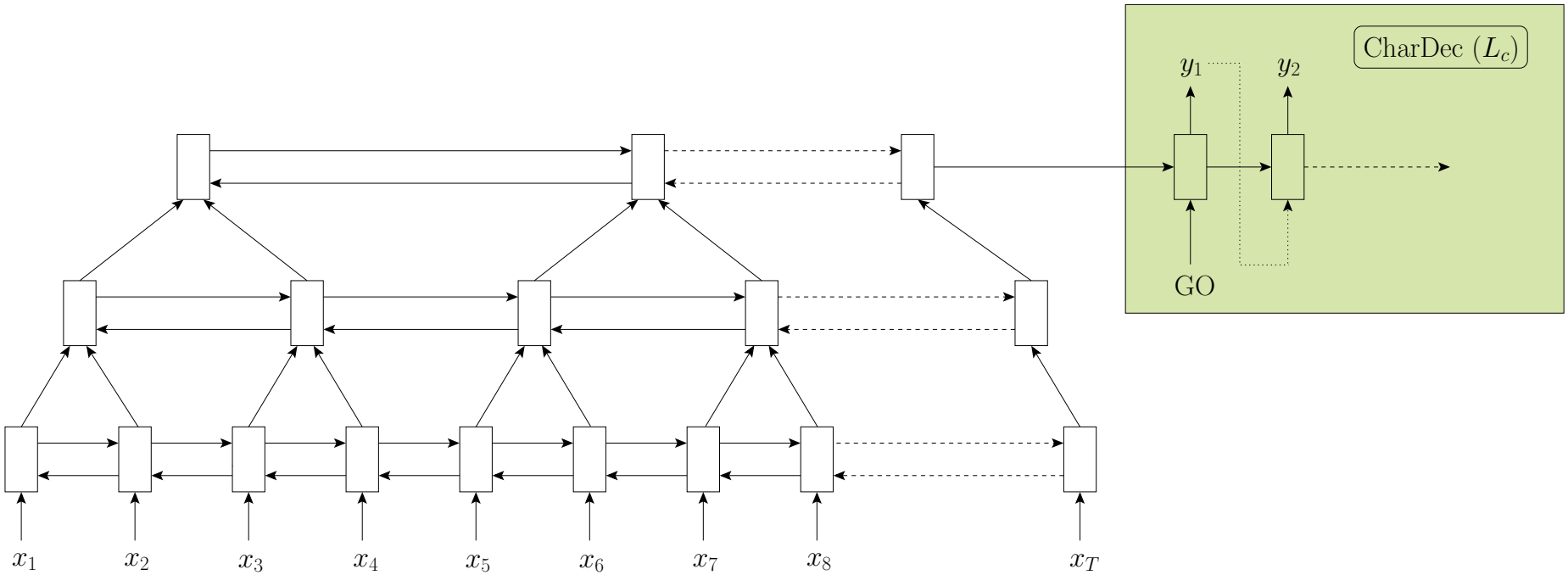


Baseline encoder-decoder attention model [Chan+ 2015]



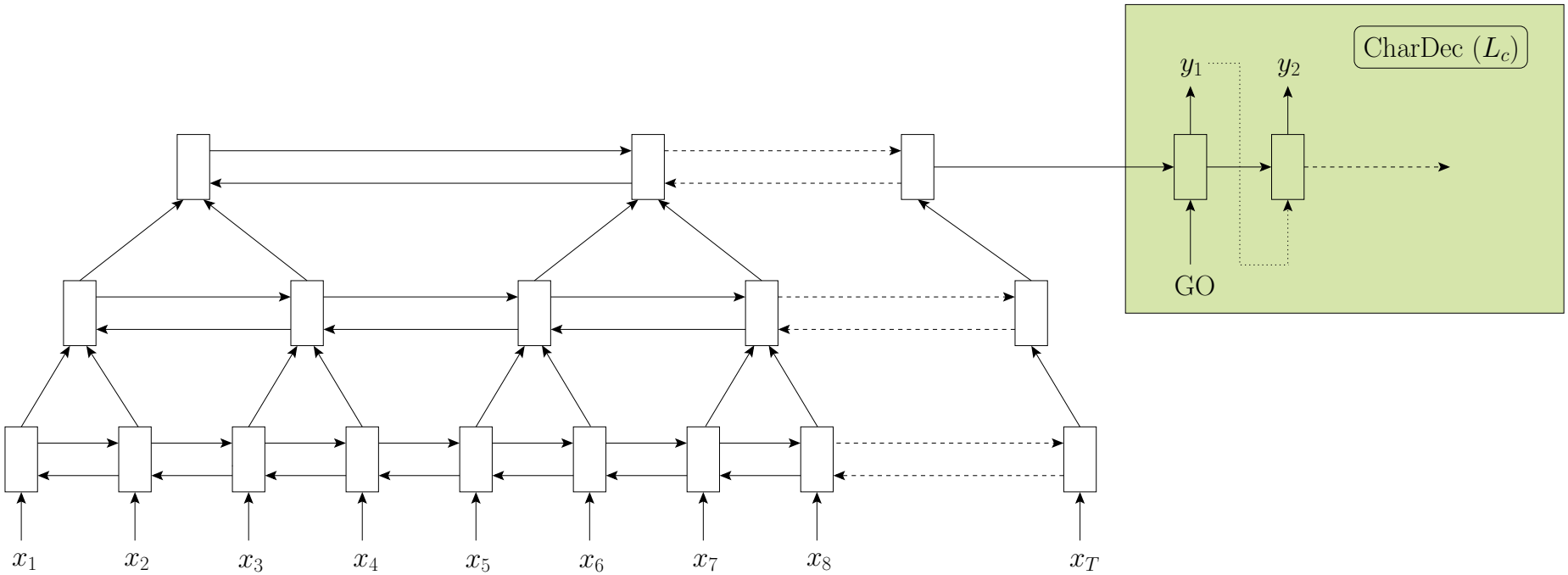
- *Speech encoder*: Pyramidal bidirectional RNN that
  - (i) Reads in spectral feature vectors $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$
  - (ii) Outputs a sequence of high-level representation vectors (hidden states)

Baseline encoder-decoder attention model [Chan+ 2015]



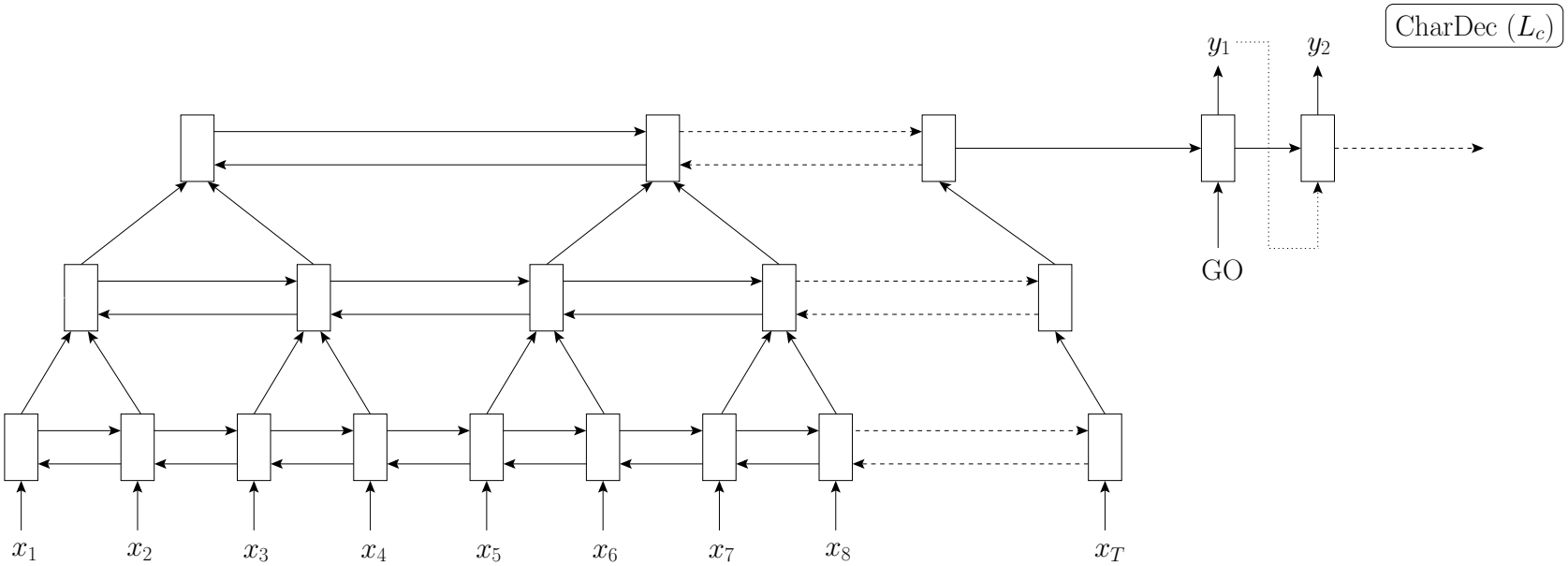
- *Speech encoder*: Pyramidal bidirectional RNN that
  - (i) Reads in spectral feature vectors $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$
  - (ii) Outputs a sequence of high-level representation vectors (hidden states)
- *Character decoder*: Unidirectional RNN that
  - (i) Uses an attention model to summarize encoder states
  - (ii) Outputs a sequence of characters $\mathbf{y} = (y_1, \ldots, y_K)$

Baseline encoder-decoder attention model [Chan+ 2015]

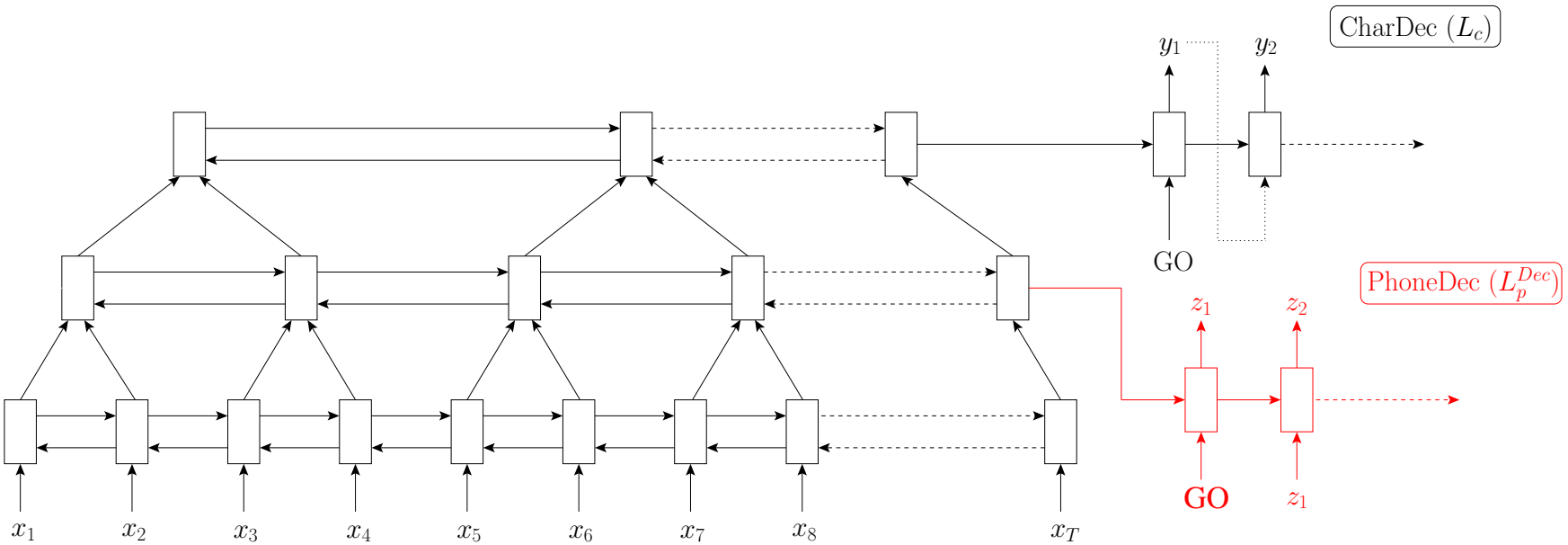


- *Speech encoder*: Pyramidal bidirectional RNN that
  - (i) Reads in spectral feature vectors $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$
  - (ii) Outputs a sequence of high-level representation vectors (hidden states)
- *Character decoder*: Unidirectional RNN that
  - (i) Uses an attention model to summarize encoder states
  - (ii) Outputs a sequence of characters $\mathbf{y} = (y_1, \ldots, y_K)$
- *Training*: Optimizes character log loss L_c

Adding phoneme-level supervision

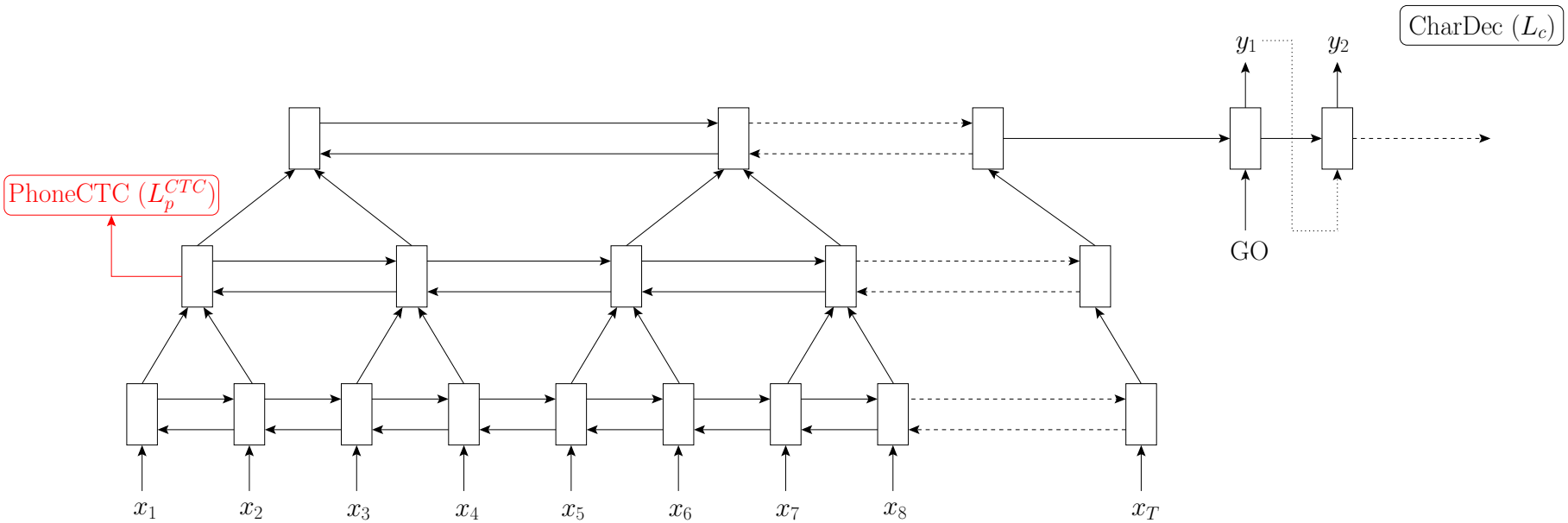


Adding phoneme-level supervision



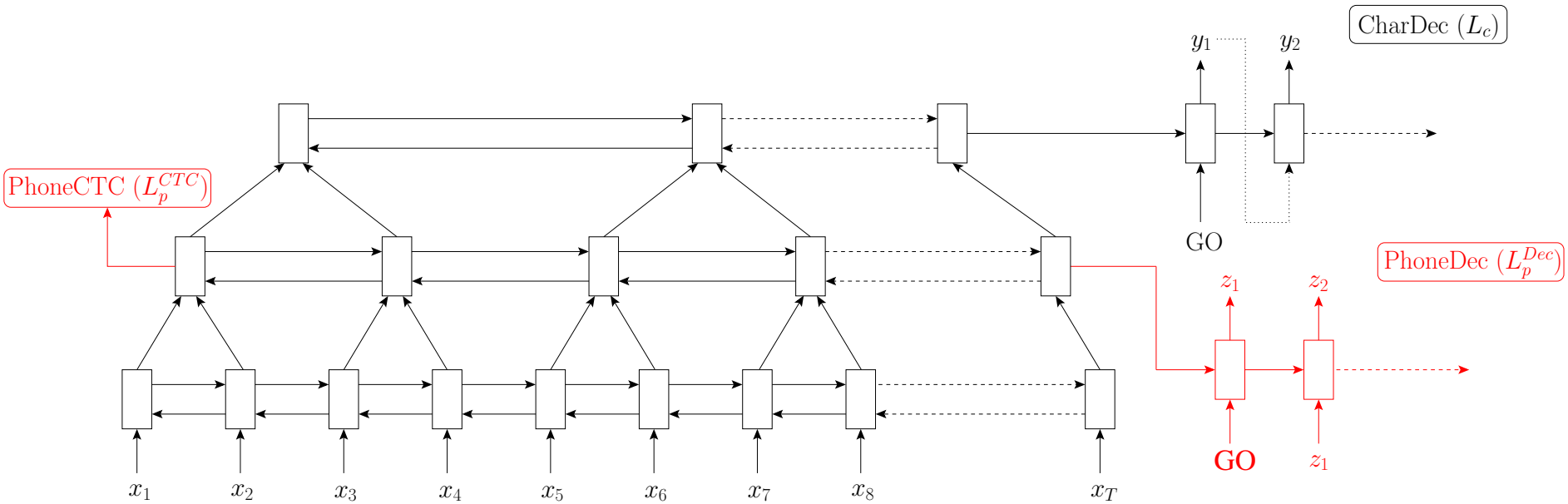
- Phoneme sequences obtained from a pronunciation dictionary

Adding phoneme-level supervision



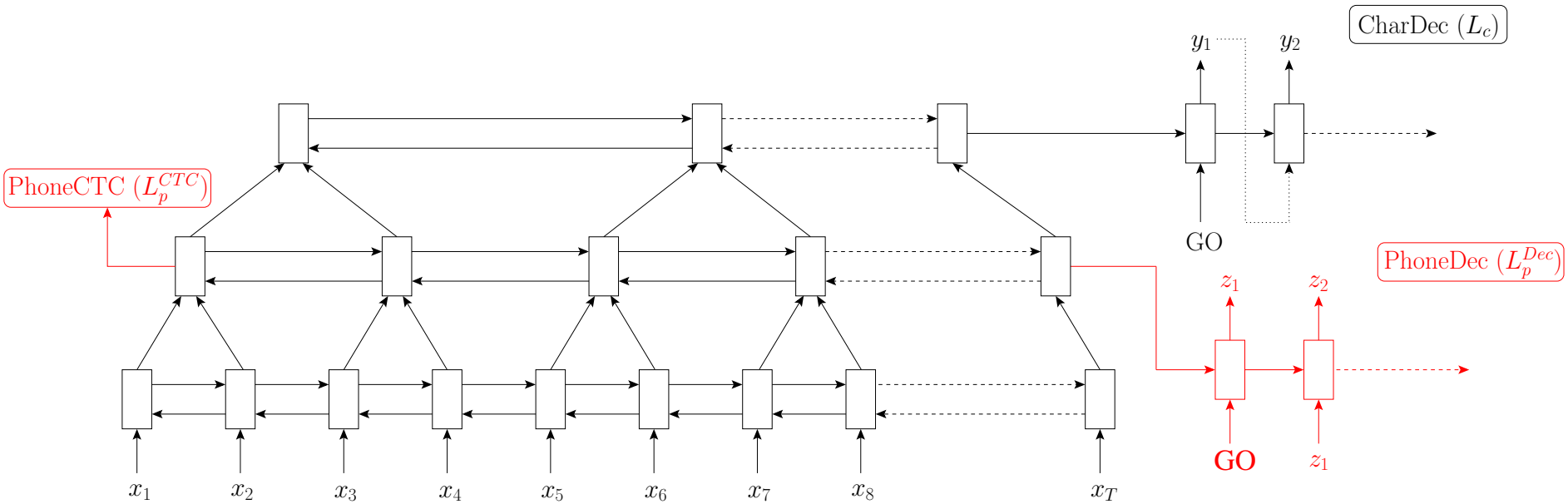
- Phoneme sequences obtained from a pronunciation dictionary
- Two types of phoneme loss:
  (a) Phoneme decoder log loss (L_p^{Dec}),

Adding phoneme-level supervision



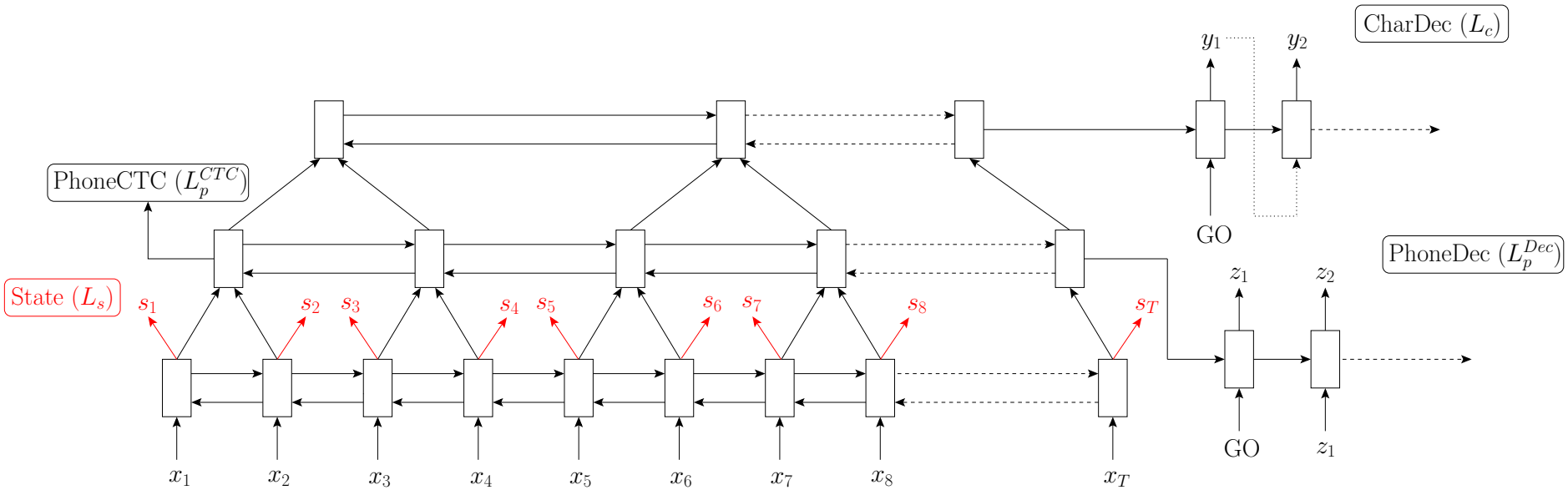
- Phoneme sequences obtained from a pronunciation dictionary
- Two types of phoneme loss:
  (a) Phoneme decoder log loss (L_p^{Dec}),
  (b) Softmax layer + CTC loss (L_p^{CTC})

Adding phoneme-level supervision



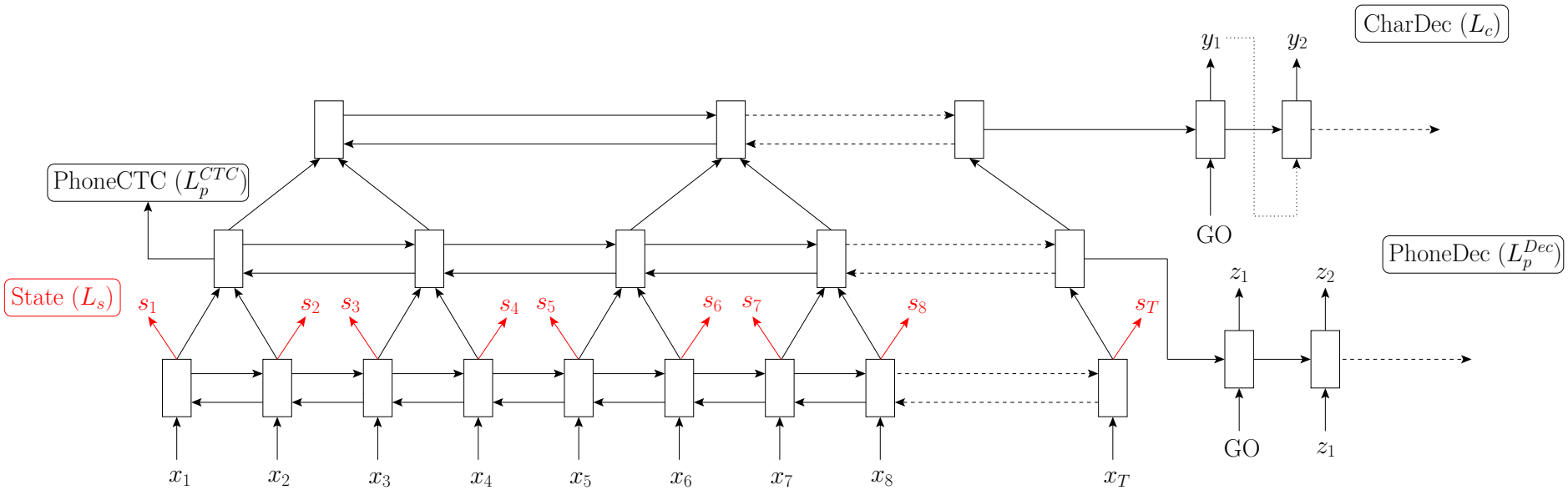
- Phoneme sequences obtained from a pronunciation dictionary
- Two types of phoneme loss:
  (a) Phoneme decoder log loss (L_p^{Dec}),
  (b) Softmax layer + CTC loss (L_p^{CTC})
- Overall training loss: $L = \frac{1}{2}(L_c + L_p)$.

Adding state-level supervision



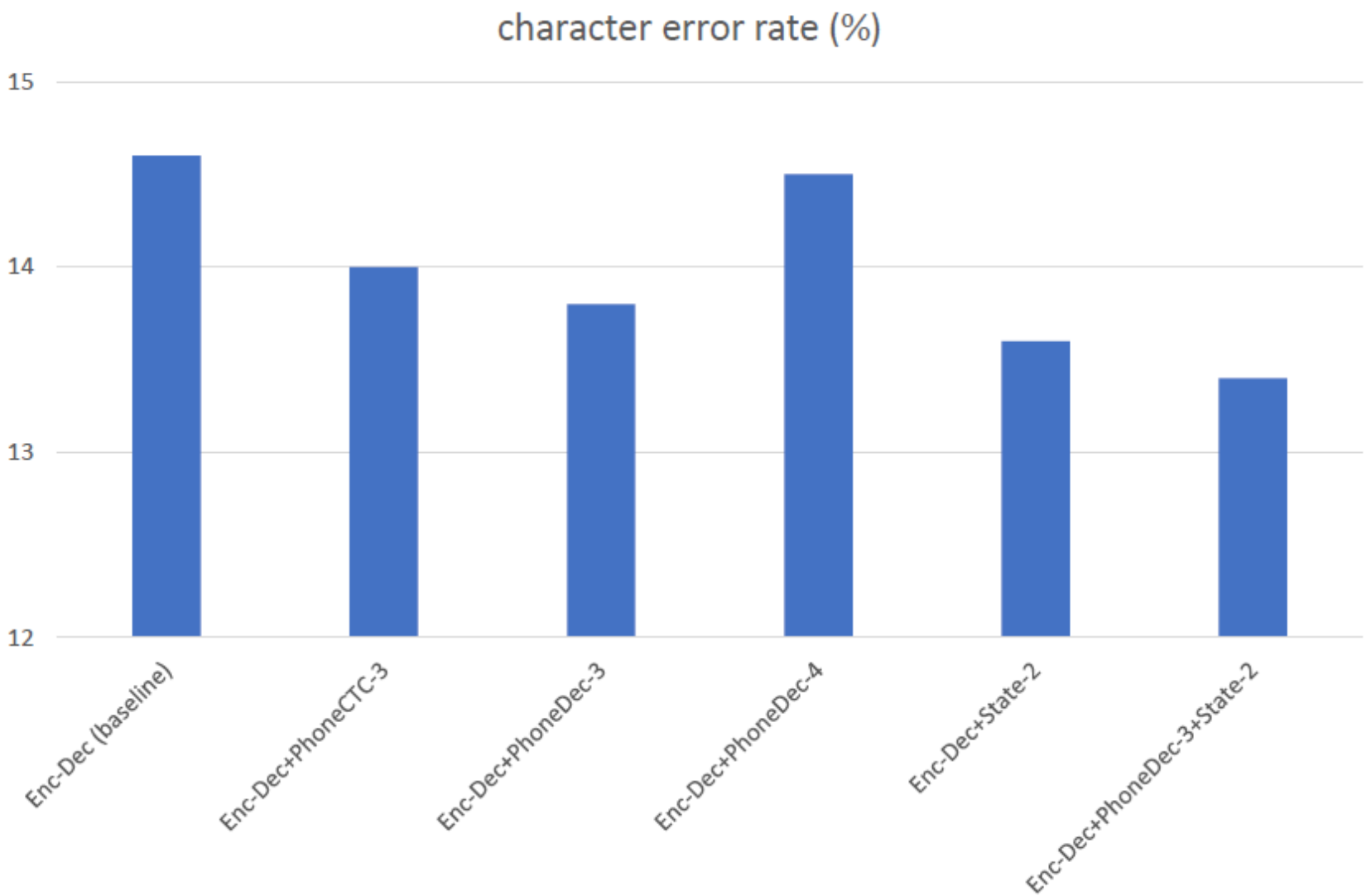
- Frame-level state labels from HMM-based alignments
- Softmax layer + state log loss
- Overall training loss: $L = \frac{1}{3}(L_c + L_p + L_s)$

Adding state-level supervision

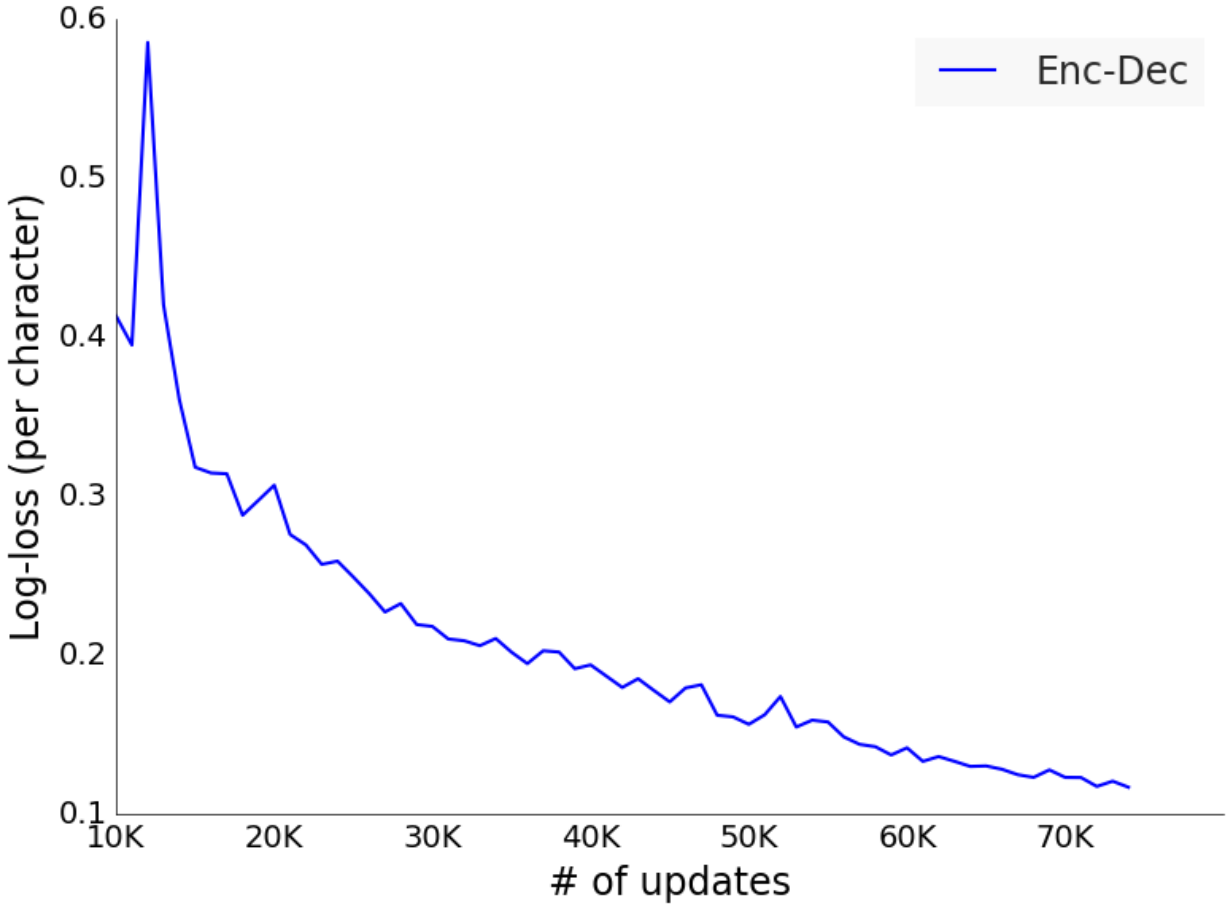


- Frame-level state labels from HMM-based alignments
- Softmax layer + state log loss
- Overall training loss: $L = \frac{1}{3}(L_c + L_p + L_s)$
- **At test time, only character decoder is used**

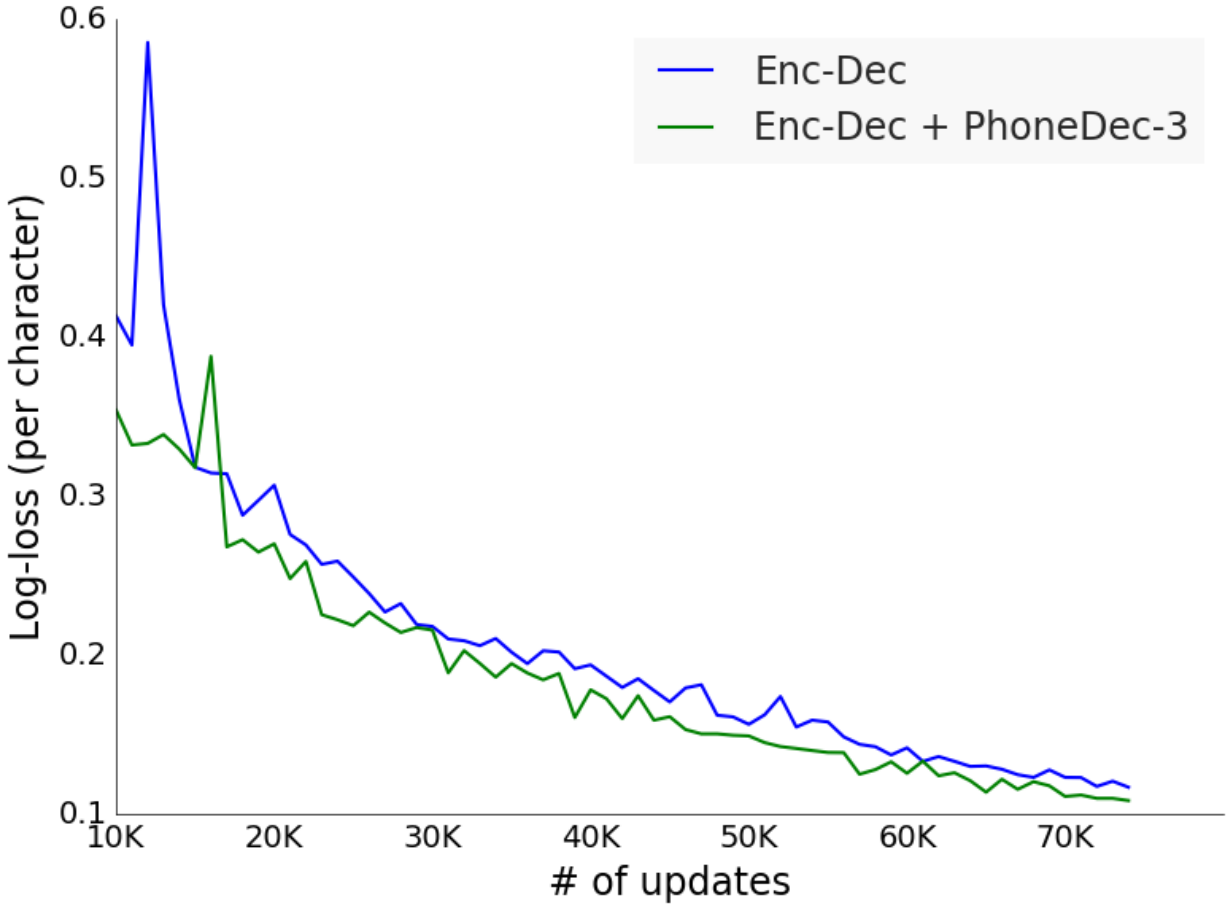
Conversational speech recognition results



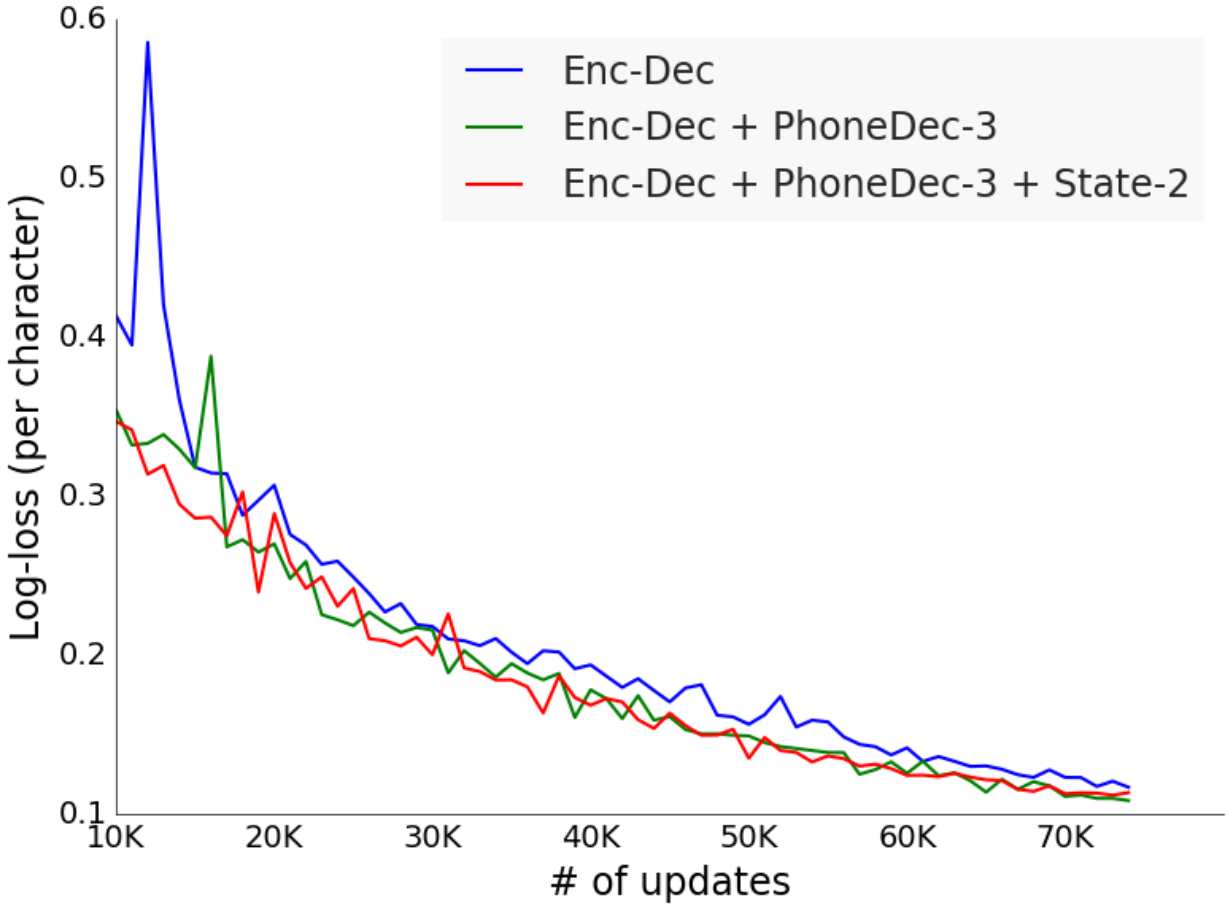
Why does multitask learning help?



Why does multitask learning help?



Why does multitask learning help?



Idea 1 summary

- Multitask learning with low-level supervision "gently" reincorporates domain knowledge into end-to-end neural learning

Idea 1 summary

- Multitask learning with low-level supervision "gently" reincorporates domain knowledge into end-to-end neural learning
- Ongoing/future work
  - Other lower-level supervision: syllables, articulatory variables, ...
  - Higher-level (semantic/syntactic) tasks

Idea 2: A multi-view approach

Joint work with Qingming Tang, Weiran Wang, Raman Arora, Galen Andrew, Jeff Bilmes, ... [ICML 2013, ICML 2015, ICASSP 2015, Interspeech 2017]

Idea 2: A multi-view approach

Joint work with Qingming Tang, Weiran Wang, Raman Arora, Galen Andrew, Jeff Bilmes, ... [ICML 2013, ICML 2015, ICASSP 2015, Interspeech 2017]

Idea: Use some other data as a proxy for domain knowledge

Idea 2: A multi-view approach

Joint work with Qingming Tang, Weiran Wang, Raman Arora, Galen Andrew, Jeff Bilmes, ... [ICML 2013, ICML 2015, ICASSP 2015, Interspeech 2017]

Idea: Use some other data as a proxy for domain knowledge

- Suppose we can't get more labeled speech, but we have access to some data paired with another type of measurement (video, physiological measurements, ...)

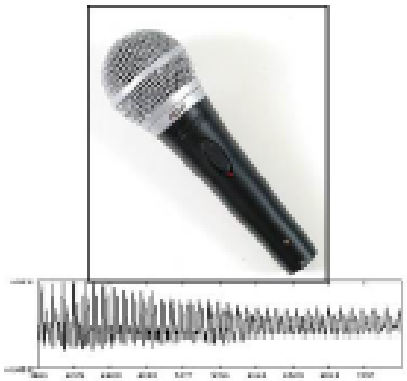
Idea 2: A multi-view approach

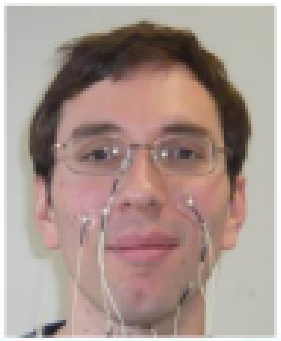
Joint work with Qingming Tang, Weiran Wang, Raman Arora, Galen Andrew, Jeff Bilmes, ... [ICML 2013, ICML 2015, ICASSP 2015, Interspeech 2017]

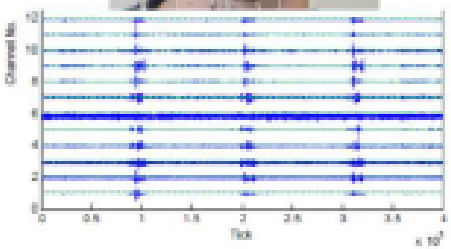
Idea: Use some other data as a proxy for domain knowledge

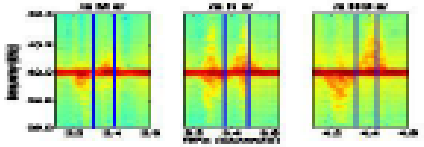
- Suppose we can't get more labeled speech, but we have access to some data paired with another type of measurement (video, physiological measurements, ...)
- The additional type of measurement ("view") is closer to the "knowledge" we seek

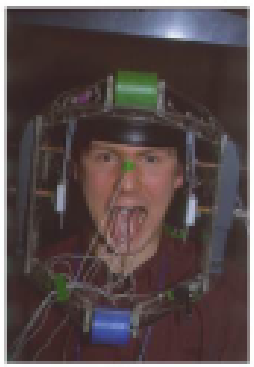
Multi-view data for speech

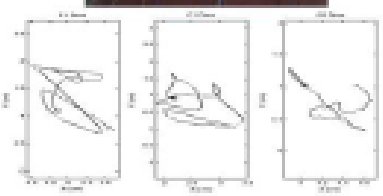


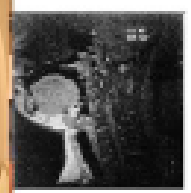








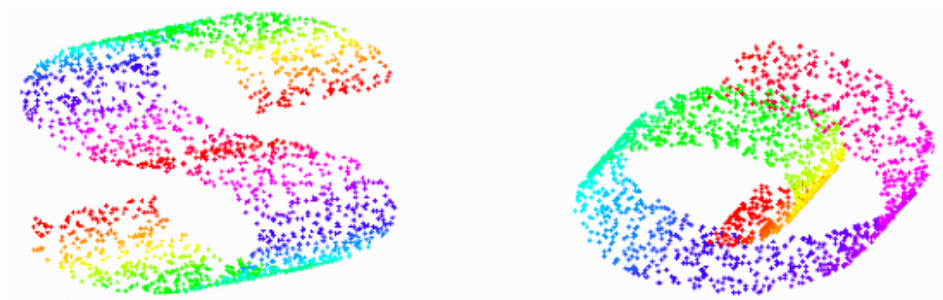




Multi-view representation learning

Training data consists of samples of a d-dimensional random vector that has some natural split into two sub-vectors:

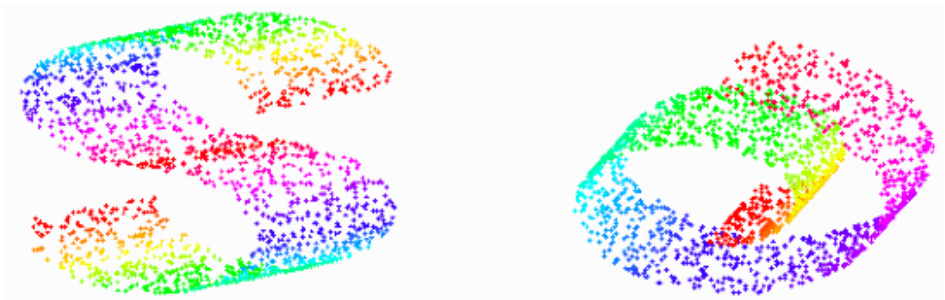
$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}, \ \mathbf{x} \in \mathbb{R}^{d_x}, \ \mathbf{y} \in \mathbb{R}^{d_y}, \ d_x + d_y = d$$



Multi-view representation learning

Training data consists of samples of a d-dimensional random vector that has some natural split into two sub-vectors:

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}, \ \mathbf{x} \in \mathbb{R}^{d_x}, \ \mathbf{y} \in \mathbb{R}^{d_y}, \ d_x + d_y = d$$

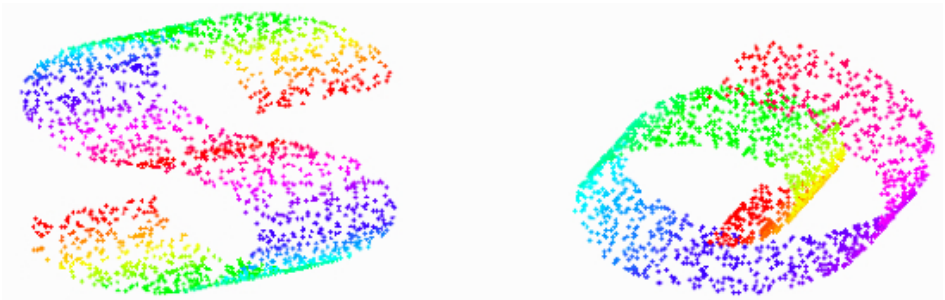


- **Multi-view representation learning**: Typically involves learning representations of each view that are **predictive** of the other, or that are **common** to both

Multi-view representation learning

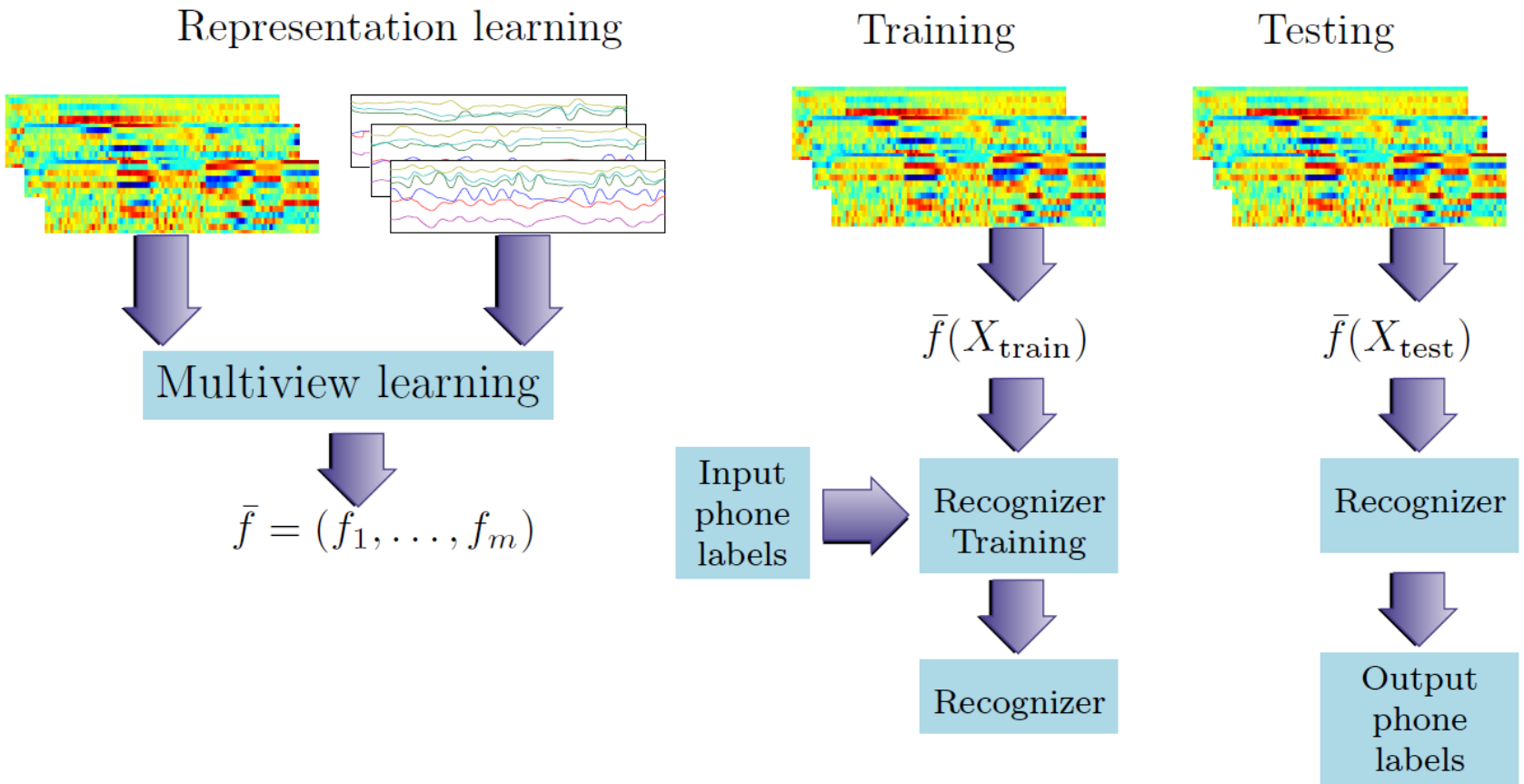
Training data consists of samples of a d-dimensional random vector that has some natural split into two sub-vectors:

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}, \ \mathbf{x} \in \mathbb{R}^{d_x}, \ \mathbf{y} \in \mathbb{R}^{d_y}, \ d_x + d_y = d$$



- **Multi-view representation learning**: Typically involves learning representations of each view that are **predictive** of the other, or that are **common** to both
- Intuition: If the noise/nuisance parameters in the two views are independent, then the shared information must be signal!

Our setup


Representation learning
Training
Testing
Multiview learning
$\bar{f} = (f_1, \ldots, f_m)$
$\bar{f}(X_{\text{train}})$
$\bar{f}(X_{\text{test}})$
Input phone labels
Recognizer Training
Recognizer
Recognizer
Output phone labels

Canonical correlation analysis (CCA)

[Hotelling 1936]

- **Given**: data set of n paired vectors $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, which are samples of random vectors $X \in \mathbb{R}^{d_x}, Y \in \mathbb{R}^{d_y}$
- **Find**: direction vectors v, w that maximize the correlation between the projections $v^T X$ and $w^T Y$

Canonical correlation analysis (CCA)

[Hotelling 1936]

- **Given**: data set of n paired vectors $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, which are samples of random vectors $X \in \mathbb{R}^{d_x}, Y \in \mathbb{R}^{d_y}$
- **Find**: direction vectors v, w that maximize the correlation between the projections $v^T X$ and $w^T Y$
- First pair of directions:

$$v_1, w_1 \quad = \quad \arg\max_{v,w} \ \mathrm{corr}(v^T X, w^T Y)$$

Canonical correlation analysis (CCA)

[Hotelling 1936]

- **Given**: data set of n paired vectors $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, which are samples of random vectors $X \in \mathbb{R}^{d_x}, Y \in \mathbb{R}^{d_y}$
- **Find**: direction vectors v, w that maximize the correlation between the projections $v^T X$ and $w^T Y$
- First pair of directions:

$$\begin{aligned} v_1, w_1 &= \arg\max_{v,w} \ \mathrm{corr}(v^T X, w^T Y) \\ &= \arg\max_{v,w} \frac{v^T C_{xy} w}{\sqrt{(v^T C_{xx} v)(w^T C_{yy} w)}} \end{aligned}$$

Canonical correlation analysis (CCA)

[Hotelling 1936]

- **Given**: data set of n paired vectors $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, which are samples of random vectors $X \in \mathbb{R}^{d_x}, Y \in \mathbb{R}^{d_y}$
- **Find**: direction vectors v, w that maximize the correlation between the projections $v^T X$ and $w^T Y$
- First pair of directions:

$$\begin{aligned} v_1, w_1 &= \arg\max_{v,w} \ \mathrm{corr}(v^T X, w^T Y) \\ &= \arg\max_{v,w} \frac{v^T C_{xy} w}{\sqrt{(v^T C_{xx} v)(w^T C_{yy} w)}} \end{aligned}$$

- Subsequent direction vectors maximize the same correlation, subject to being uncorrelated with the previous directions

Canonical correlation analysis (CCA)

Solution can be found via an eigenproblem:

$$\begin{aligned} C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx} v &= \lambda^2 v \\ w &\propto C_{yy}^{-1} C_{yx} v \end{aligned}$$

Canonical correlation analysis (CCA)

Solution can be found via an eigenproblem:

$$\begin{aligned} C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx} v &= \lambda^2 v \\ w &\propto C_{yy}^{-1} C_{yx} v \end{aligned}$$

Canonical correlation analysis (CCA)

Solution can be found via an eigenproblem:

$$\begin{aligned} C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx} v &= \lambda^2 v \\ w &\propto C_{yy}^{-1} C_{yx} v \end{aligned}$$

We use CCA for dimensionality reduction of View 1 (X):

- Compute first k CCA directions
- Project new data from View 1 onto those directions

Canonical correlation analysis (CCA)

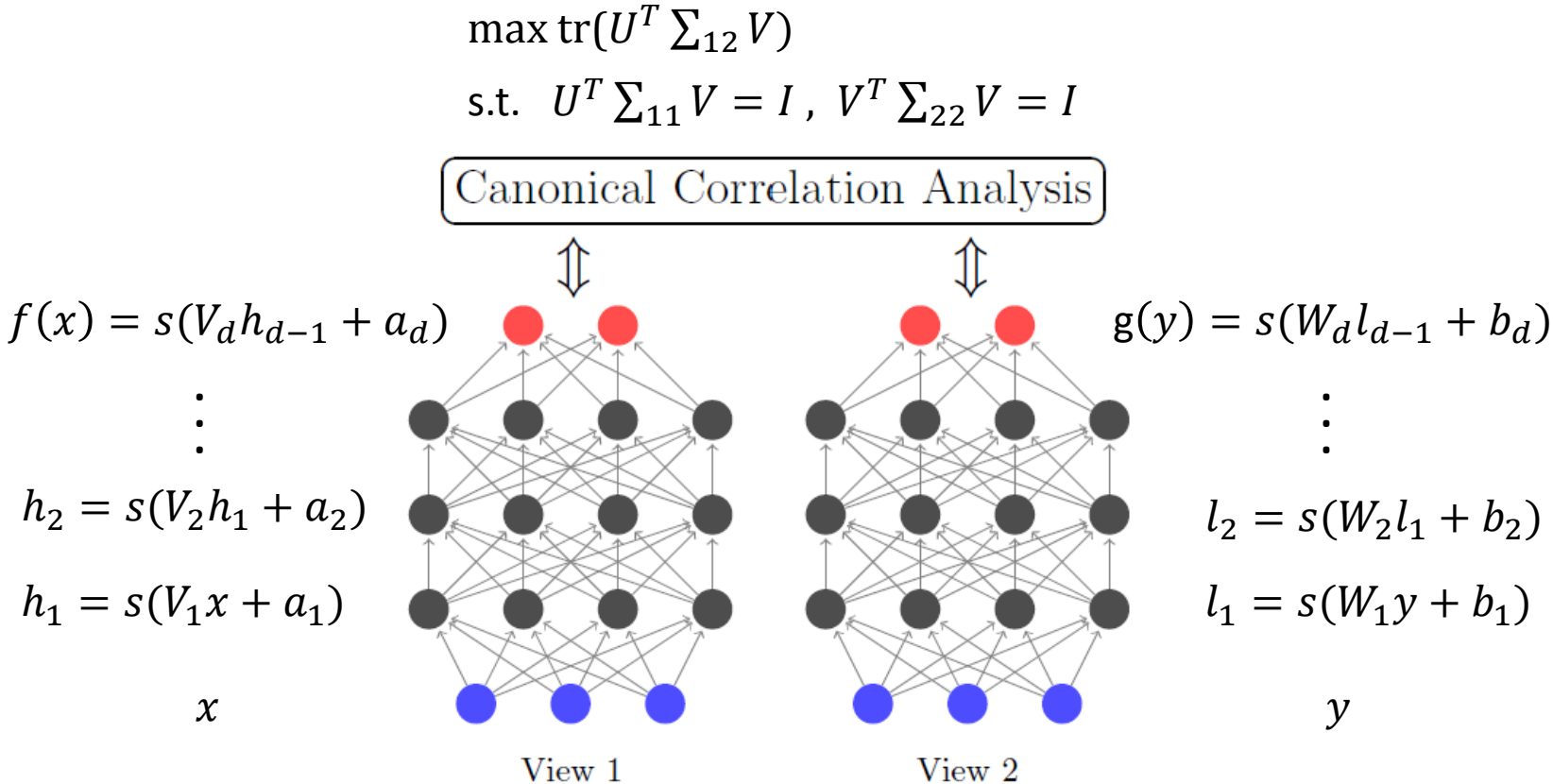
Alternative formulation: Given paired data matrices X and Y, find $V \in \mathbb{R}^{d_x \times k}, W \in \mathbb{R}^{d_y \times k}$ that

$$\begin{aligned} \text{minimize :} \quad & ||W^T Y - V^T X||_F \\ \text{subject to :} \quad & V^T C_{xx} V = W^T C_{yy} W = I \end{aligned}$$

Deep CCA [Andrew+ 2013]

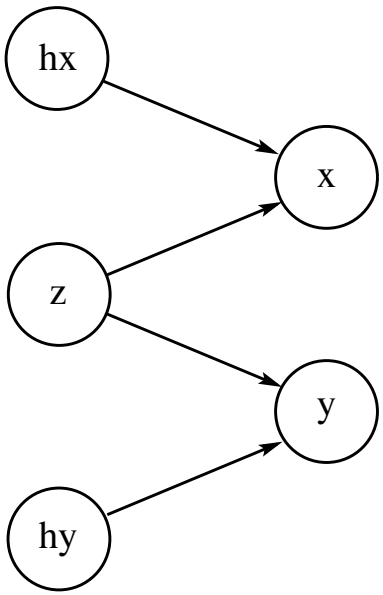
Deep CCA [Andrew+ 2013]

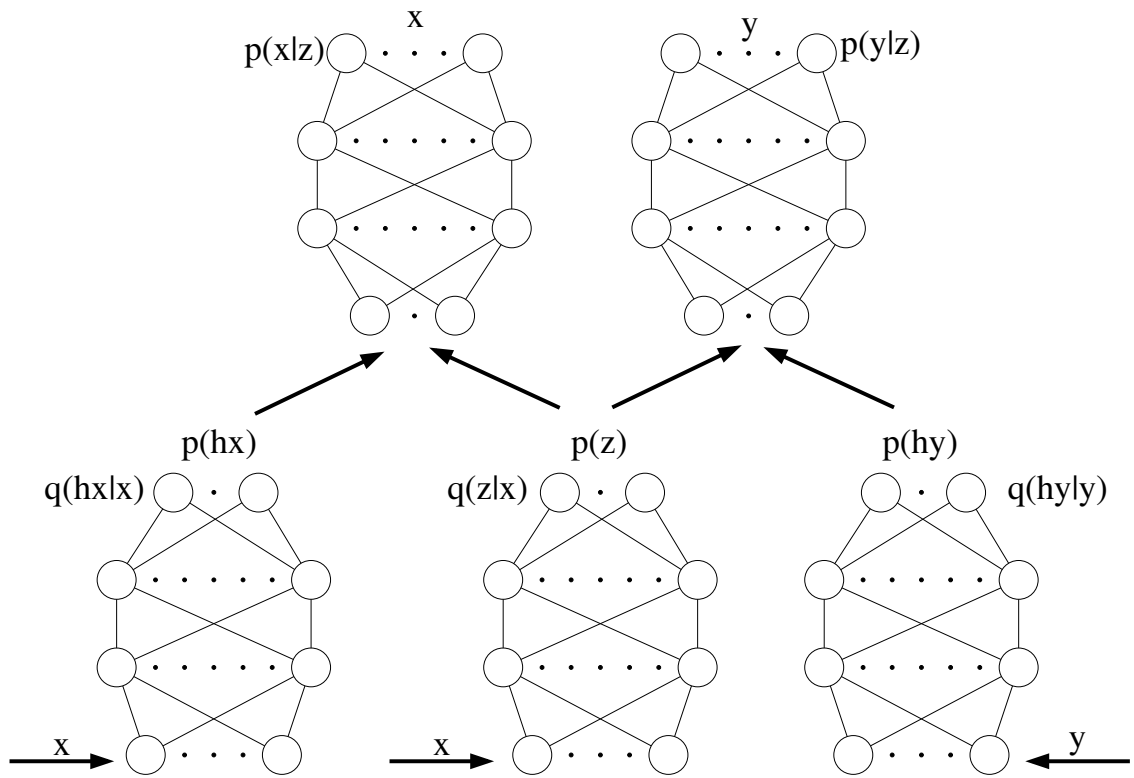
- Nonlinear extension of CCA
- Each view's representation is the output of a neural network
- All parameters learned jointly via backpropagation



Deep variational CCA [Wang+ 2016, Tang+ 2017]

- Inspired by generative interpretaion of CCA [Bach & Jordan 2005]
- Loosely, like a multi-view extension of variational autoencoders





Contrastive loss [Hermann & Blunsom 2014]

Contrastive loss [Hermann & Blunsom 2014]

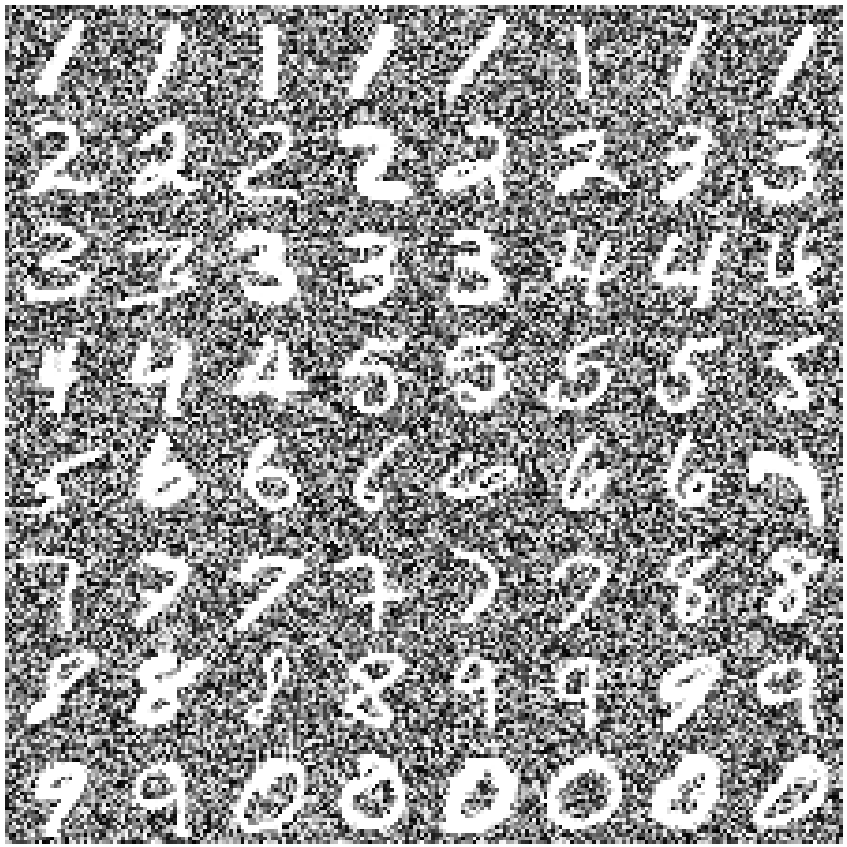
Alternative to CCA

- Try to bring paired examples closer together
- While keeping random unpaired examples farther apart by some margin

$$\min_{f,g} \frac{1}{N} \sum_{i=1}^{N} \max\left(0, m + dist(f(x_i^+), g(y_i^+)) - dist(f(x_i^+), g(y_i^-))\right)$$

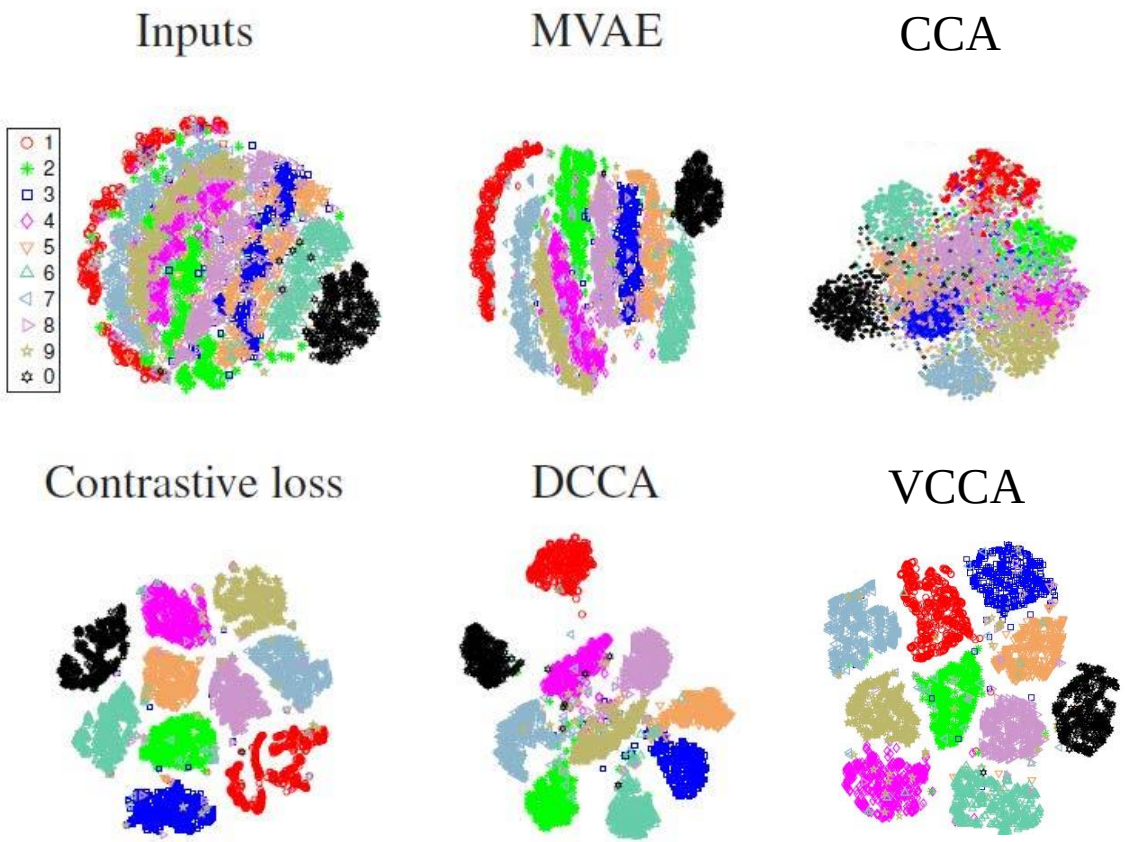
Simulated example: Noisy MNIST digits

A simulated dataset that perfectly satisfies the uncorrelated noise multi-view assumption



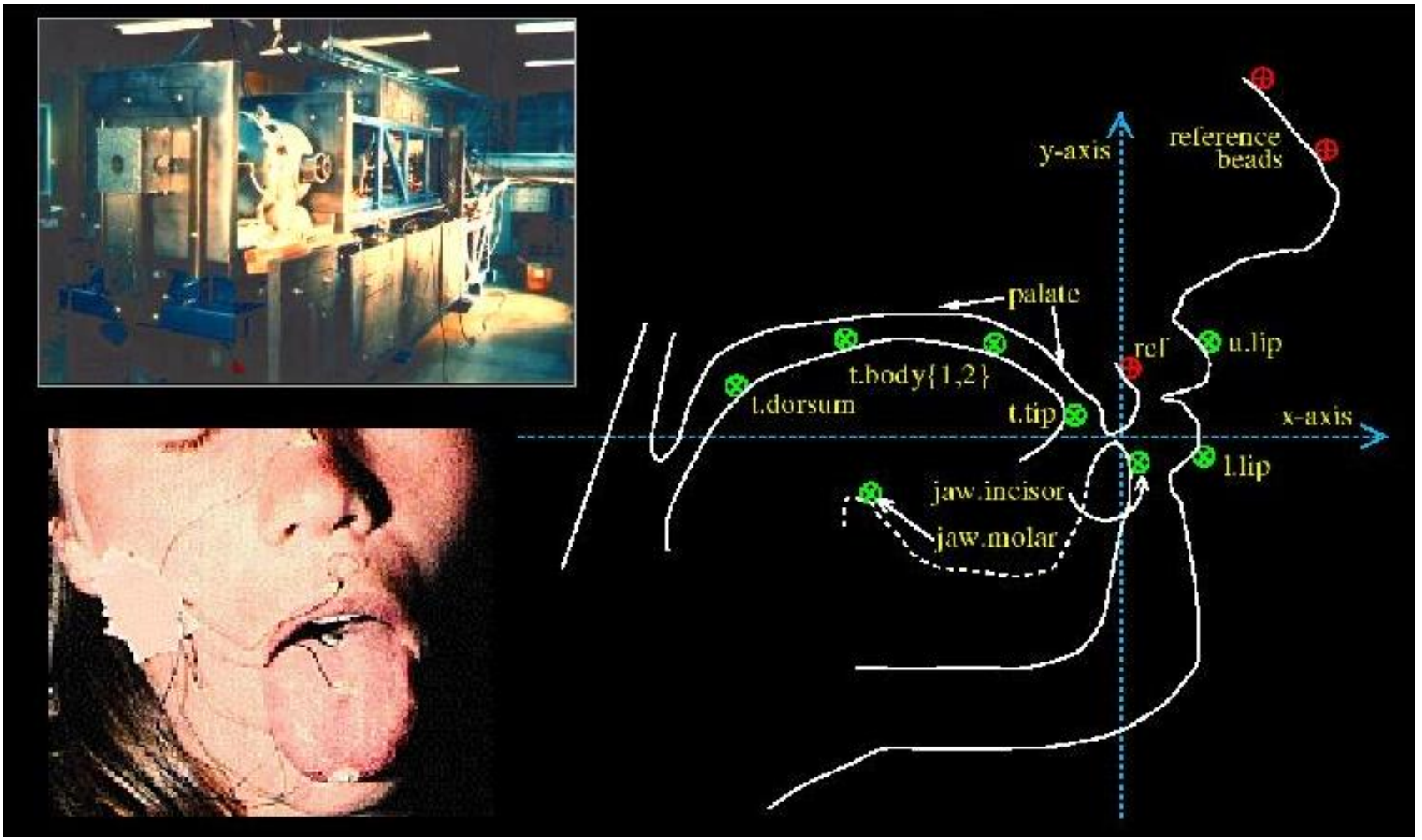
Noisy MNIST visualization [Wang+ 2015, Wang+ 2016]

Visualization via t-SNE [van der Maaten & Hinton 2008]



Speech recognition experiments

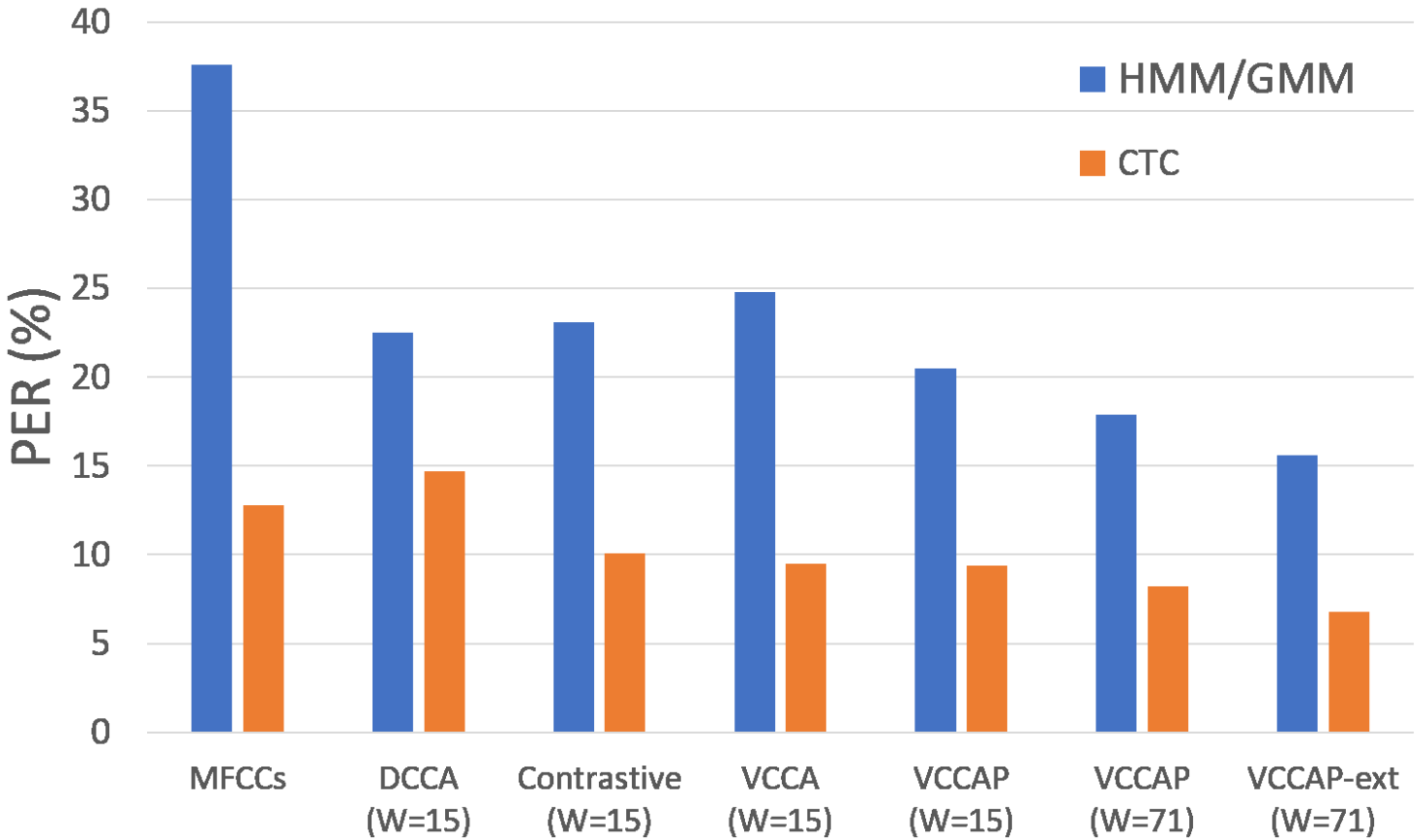
U. Wisconsin X-ray Microbeam Database (XRMB) [Westbury+ 1994]



Speech recognition experiments

- Acoustic view: MFCCs concatenated over W frames
- Articulatory view: on horizontal + vertical displacements of 8 pellets, concatenated over W frames
- 35 speakers for representation learning, 12 for recognition experiments

Speech recognition results [Tang+ 2017]



Idea 2 summary

- Multi-view representation learning uses another data view as a proxy for domain knowledge

Idea 2 summary

- Multi-view representation learning uses another data view as a proxy for domain knowledge
- Ongoing work
  - Domain-independent multi-view representations
  - Combining with the multitask approach: Multi-view representation loss can be viewed as an additional low-level "task"

Idea 2 summary

- Multi-view representation learning uses another data view as a proxy for domain knowledge
- Ongoing work
  - Domain-independent multi-view representations
  - Combining with the multitask approach: Multi-view representation loss can be viewed as an additional low-level "task"
  - (Aside: Similar methods apply to image+caption representations [Wang+ 2016], multi-lingual word embeddings [Lu+ 2015])

Discussion

Domain knowledge is less critical than it has been in the past for certain data-rich tasks (this is good!)

- But it can help in low-data settings or when optimization is hard
- Two ideas:
  - Multitask learning with low-level auxiliary tasks: Low-level tasks encourage good intermediate representations
  - Multi-view representation learning: 2nd view as a proxy for knowledge

Other important settings

- A little annotated data, a lot of unlabeled data $\longrightarrow$ unsupervised and semisupervised approaches
- Annotated data available, but for a different task $\longrightarrow$ transfer learning