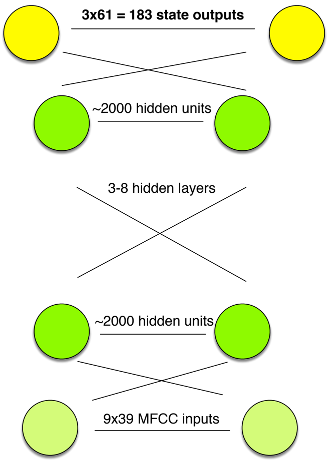
Neural Networks for Acoustic Modelling 3: Context-dependent DNNs and TDNNs

Steve Renals

Automatic Speech Recognition – ASR Lecture 9
11 February 2019

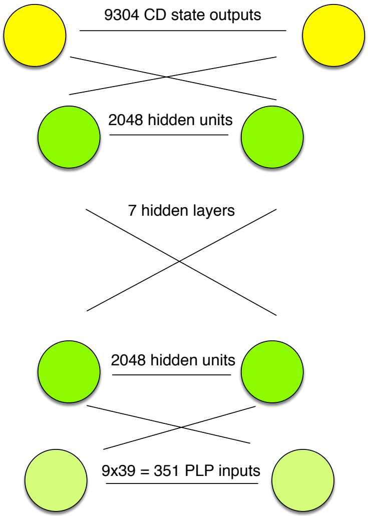
Recap: DNN for TIMIT



- **Deeper**: Deep neural network architecture – multiple hidden layers
- **Wider**: Use HMM state alignment as outputs rather than hand-labelled phones – 3-state HMMs, so 3×48 states
- Training many hidden layers is computationally expensive – use GPUs to provide the computational power

Context Dependent DNN Acoustic Models

DNN acoustic model for Switchboard



(Hinton et al (2012))

Context-dependent hybrid HMM/DNN

- First train a context-dependent HMM/GMM system on the same data, using a phonetic decision tree to determine the HMM tied states
- Perform Viterbi alignment using the trained HMM/GMM and the training data
- Train a neural network to map the input speech features to a label representing a context-dependent tied HMM state
  - So the size of the label set is thousands (number of context-dependent tied states) rather than tens (number of context-independent phones) Each frame is labelled with the Viterbi aligned tied state
- Train the neural network using gradient descent as usual
- Use the context-dependent scaled likelihoods obtained from the neural network when decoding

Example: hybrid HMM/DNN large vocabulary conversational speech recognition (Switchboard)

- Recognition of American English conversational telephone speech (Switchboard)
- Baseline context-dependent HMM/GMM system
  - 9,304 tied states
  - Discriminatively trained (BMMI — similar to MPE)
  - 39-dimension PLP (+ derivatives) features
  - Trained on 309 hours of speech
- Hybrid HMM/DNN system
  - Context-dependent — 9304 output units obtained from Viterbi alignment of HMM/GMM system
  - 7 hidden layers, 2048 units per layer
- DNN-based system results in significant word error rate reduction compared with GMM-based system

DNN vs GMM on large vocabulary tasks (Experiments from 2012)

[TABLE 3] A COMPARISON OF THE PERCENTAGE WERs USING DNN-HMMs AND GMM-HMMs ON FIVE DIFFERENT LARGE VOCABULARY TASKS.

TASK	HOURS OF TRAINING DATA	DNN-HMM	GMM-HMM WITH SAME DATA	GMM-HMM WITH MORE DATA
SWITCHBOARD (TEST SET 1)	309	18.5	27.4	18.6 (2,000 H)
SWITCHBOARD (TEST SET 2)	309	16.1	23.6	17.1 (2,000 H)
ENGLISH BROADCAST NEWS	50	17.5	18.8	
BING VOICE SEARCH (SENTENCE ERROR RATES)	24	30.4	36.2	
GOOGLE VOICE INPUT	5,870	12.3		16.0 (>> 5,870 H)
YOUTUBE	1,400	47.6	52.3	

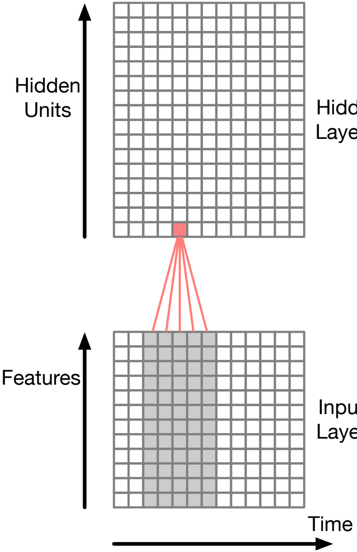
(Hinton et al (2012))

TDNNs
Time-delay Neural Networks

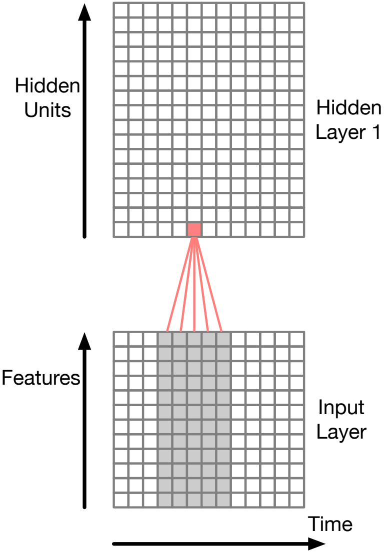
Modelling acoustic context

- DNNs allow the network to model acoustic context by including neighbouring frame in the input layer – the output is thus estimating the phone or state probability using that contextual information
- Richer NN models of acoustic context
  - **Time-delay neural networks (TDNNs)**
    - each layer processes a context window from the previous layer
    - higher hidden layers have a wider receptive field into the input
  - **Recurrent neural networks (RNNs)**
    - hidden units at time t take input from their value at time $t - 1$
    - these recurrent connections allow the network to learn state
  - Both approaches try to learn invariances in time, and form representations based on compressing the history of observations

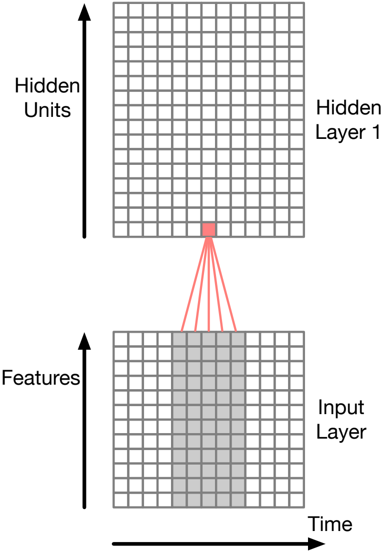
TDNNs – first hidden layer receptive field



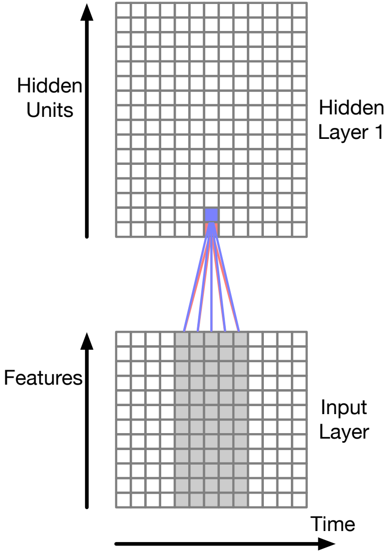
TDNNs – first hidden layer receptive field



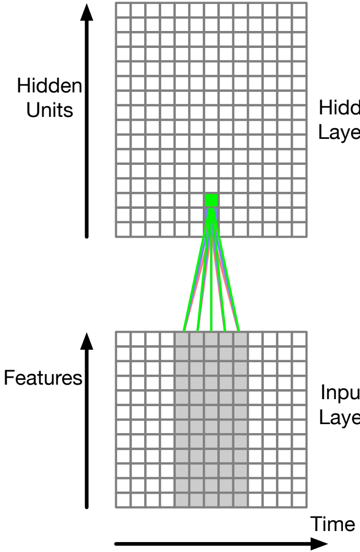
TDNNs – first hidden layer receptive field



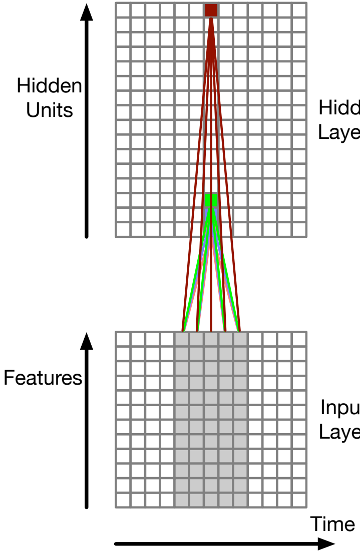
TDNNs – first hidden layer receptive field



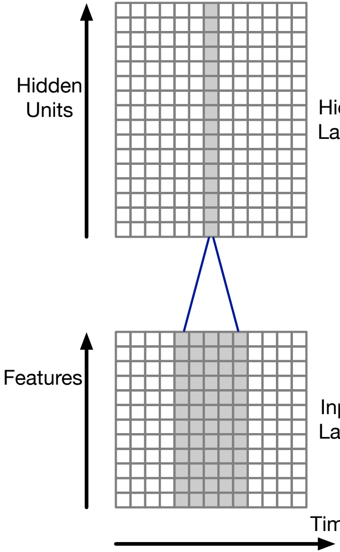
TDNNs – first hidden layer receptive field



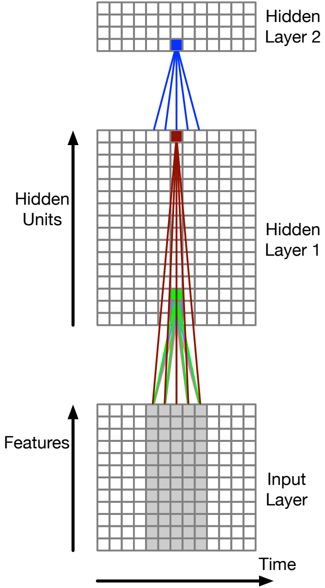
TDNNs – first hidden layer receptive field



TDNNs – first hidden layer receptive field

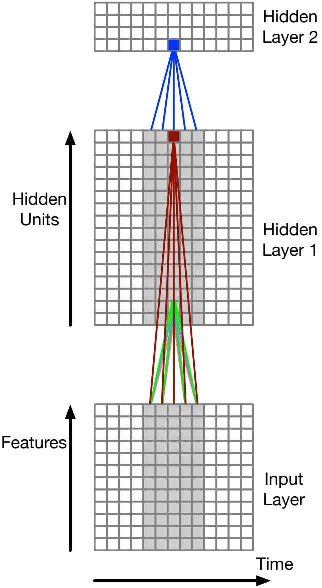


TDNNs – second hidden layer receptive field



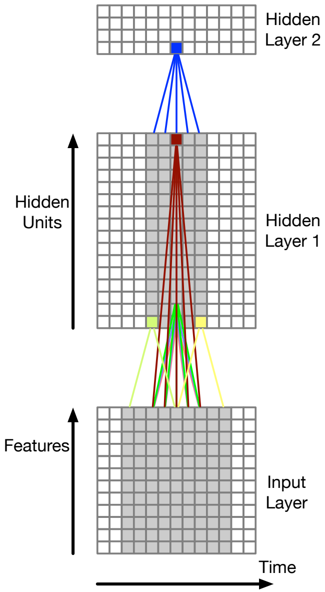
- Higher hidden layers take input from a time window over the previous hidden layer
- Lower hidden layers learn from narrower contexts, higher hidden layers from wider acoustic contexts
- Receptive field increases for higher hidden layers

TDNNs – second hidden layer receptive field



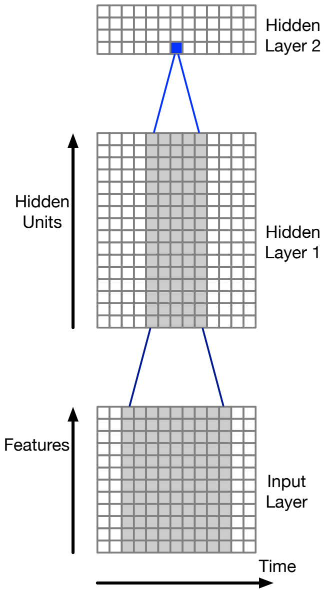
- Higher hidden layers take input from a time window over the previous hidden layer
- Lower hidden layers learn from narrower contexts, higher hidden layers from wider acoustic contexts
- Receptive field increases for higher hidden layers

TDNNs – second hidden layer receptive field



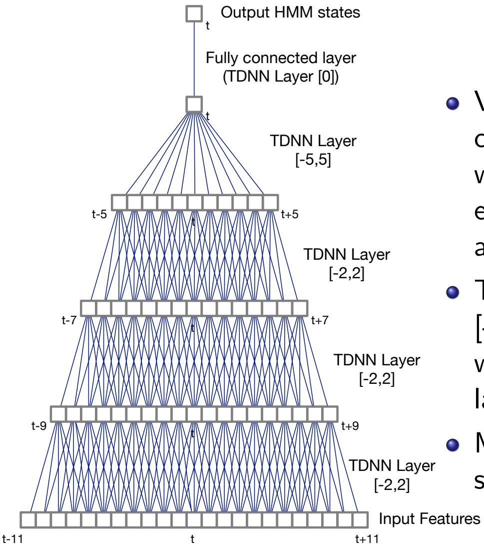
- Higher hidden layers take input from a time window over the previous hidden layer
- Lower hidden layers learn from narrower contexts, higher hidden layers from wider acoustic contexts
- Receptive field increases for higher hidden layers

TDNNs – second hidden layer receptive field



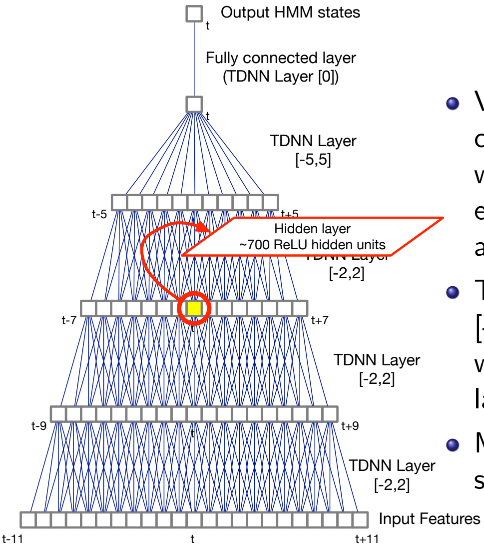
- Higher hidden layers take input from a time window over the previous hidden layer
- Lower hidden layers learn from narrower contexts, higher hidden layers from wider acoustic contexts
- Receptive field increases for higher hidden layers

Example TDNN Architecture



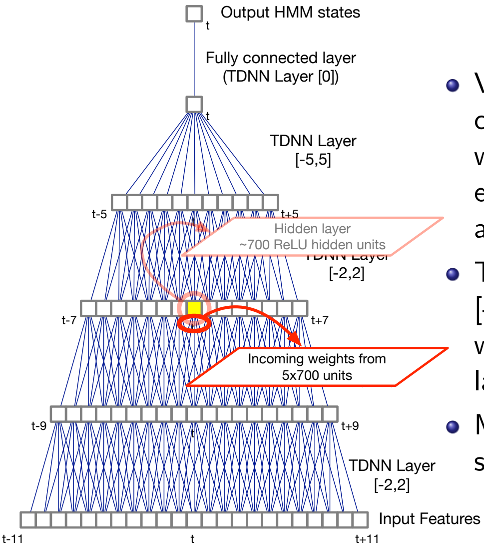
- View a TDNN as a 1D convolutional network with the transforms for each hidden unit tied across time
- TDNN layer with context [-2,2] has 5x as many weights as a regular DNN layer
- More computation, more storage required!

Example TDNN Architecture



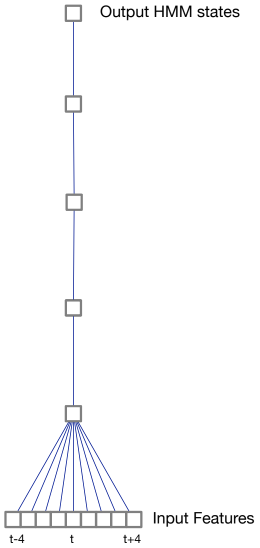
- View a TDNN as a 1D convolutional network with the transforms for each hidden unit tied across time
- TDNN layer with context [-2,2] has 5x as many weights as a regular DNN layer
- More computation, more storage required!

Example TDNN Architecture

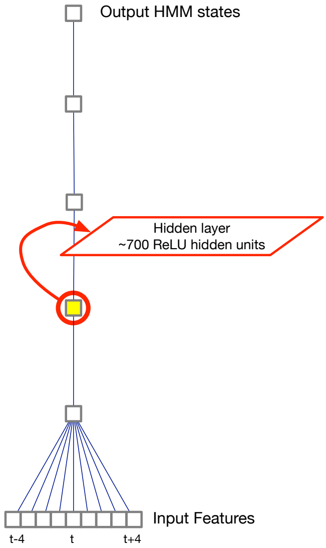


- View a TDNN as a 1D convolutional network with the transforms for each hidden unit tied across time
- TDNN layer with context [-2,2] has 5x as many weights as a regular DNN layer
- More computation, more storage required!

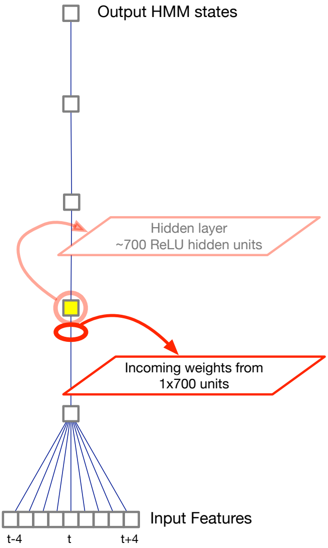
Comparison with DNN with input window



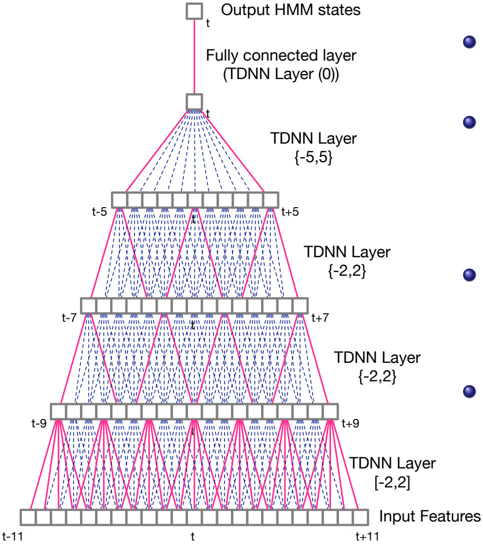
Comparison with DNN with input window



Comparison with DNN with input window

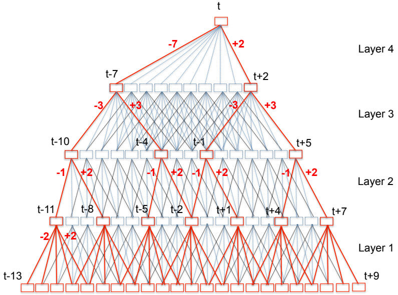


Sub-sampled TDNN



- Sub sample window of hidden unit activations
- Large overlaps between input contexts at adjacent time steps – likely to be correlated
- Allow gaps between frames in a window (cf. dilated convolutions)
- Sub-sampling saves computation and reduces number of model size (number of weights)

Example sub-sampled TDNN



Peddinti (2015)

Layer	Context	Sub-sampled Context
1	[-2,2]	[-2,2]
2	[-1,2]	{-1,2}
3	[-3,3]	{-3,3}
4	[-7,2]	{-7,2}
5	{0}	{0}

- Increase the context for higher layers of the network
- Subsampled so that difference between sampled hidden units is multiple of 3 to enable “clean” sub-sampling
- Asymmetric contexts
- MFCC features in this case

Switchboard results – DNN vs TDNN

Table 2: Performance comparison of DNN and TDNN with various temporal contexts

Model	Network Context	Layerwise Context					WER	
		1	2	3	4	5	Total	SWB
DNN-A	$[-7,7]$	$[-7,7]$	$\{0\}$	$\{0\}$	$\{0\}$	$\{0\}$	22.1	15.5
DNN-A_2	$[-7,7]$	$[-7,7]$	$\{0\}$	$\{0\}$	$\{0\}$	$\{0\}$	**21.6**	**15.1**
DNN-B	$[-13,9]$	$[-13,9]$	$\{0\}$	$\{0\}$	$\{0\}$	$\{0\}$	22.3	15.7
DNN-C	$[-16,9]$	$[-16,9]$	$\{0\}$	$\{0\}$	$\{0\}$	$\{0\}$	22.3	15.7
TDNN-A	$[-7,7]$	$[-2,2]$	$\{-2,2\}$	$\{-3,4\}$	$\{0\}$	$\{0\}$	21.2	14.6
TDNN-B	$[-9,7]$	$[-2,2]$	$\{-2,2\}$	$\{-5,3\}$	$\{0\}$	$\{0\}$	21.2	14.5
TDNN-C	$[-11,7]$	$[-2,2]$	$\{-1,1\}$	$\{-2,2\}$	$\{-6,2\}$	$\{0\}$	20.9	14.2
TDNN-D	$[-13,9]$	$[-2,2]$	$\{-1,2\}$	$\{-3,4\}$	$\{-7,2\}$	$\{0\}$	**20.8**	**14.0**
TDNN-E	$[-16,9]$	$[-2,2]$	$\{-2,2\}$	$\{-5,3\}$	$\{-7,2\}$	$\{0\}$	20.9	14.2

Peddinti (2015)

DNN vs TDNN on other datasets

Database	Size	WER		Rel. Change
		DNN	TDNN	
Res. Management	3h hrs	2.27	2.30	-1.3
Wall Street Journal	80 hrs	6.57	6.22	5.3
TedLIUM	118 hrs	19.3	17.9	7.2
Switchboard	300 hrs	15.5	14.0	9.6
Librispeech	960 hrs	5.19	4.83	6.9
Fisher English	1800 hrs	22.24	21.03	5.4

Peddinti (2015)

Summary and Conclusions

- Scaling DNNs for large vocabulary speech recognition
- Context-dependent DNNs – use state clusters from CD HMM/GMM as output labels – results in significant improvements in accuracy for DNNs over GMMs
- Richer temporal modelling – time-delay neural networks (TDNNs)
- Sub-sampled TDNNs

Reading

- A Maas et al (2017). "Building DNN acoustic models for large vocabulary speech recognition", *Computer Speech and Language*, **41**:195–213. https://web.stanford.edu/class/cs224s/papers/maas_et_al_2017.pdf
- V Peddinti et al (2015). "A time delay neural network architecture for efficient modeling of long temporal contexts", *Interspeech*. https://www.isca-speech.org/archive/interspeech_2015/i15_3214.html

Background Reading:

- G Hinton et al (Nov 2012). "Deep neural networks for acoustic modeling in speech recognition", *IEEE Signal Processing Magazine*, **29**(6), 82–97. http://ieeexplore.ieee.org/document/6296526