# Robustness is dead! Long live robustness!

Michael L. Seltzer

Microsoft Research

REVERB 2014 | May 10, 2014

Collaborators: Dong Yu, Yan Huang, Frank Seide, Jinyu Li, Jui-Ting Huang

# Golden age of speech recognition

- More investment, more languages, and more data than ever before

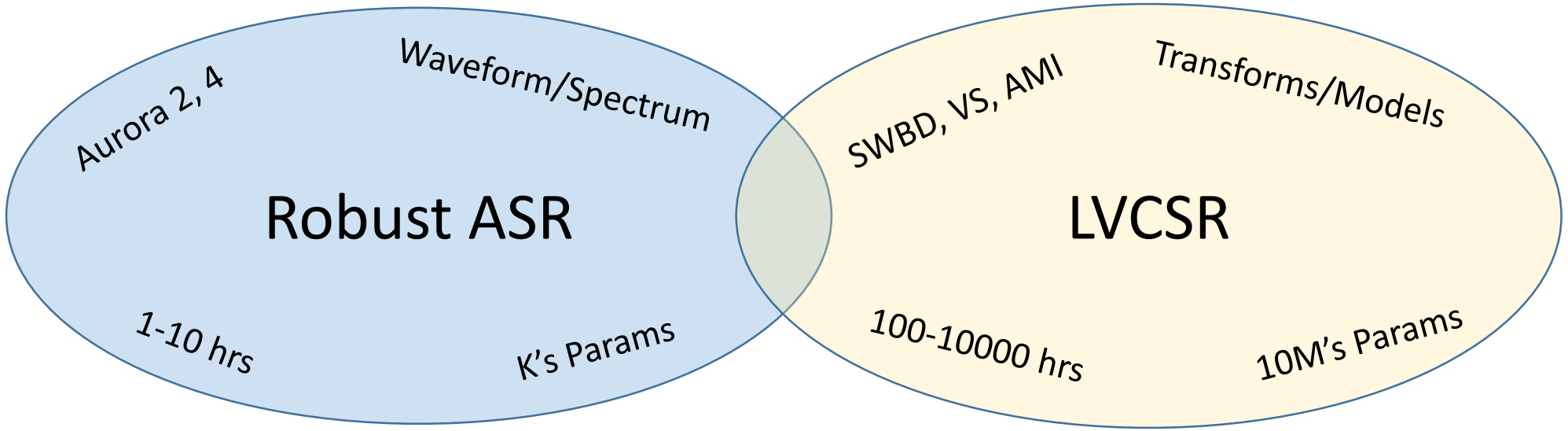
# Golden age of robustness?

- Yes!


- Many large scale deployments in challenging environments

# Golden age of robustness?

- No!




- No overlap in software, tools, systems means no common ground

# Finding common ground…

- DNNs + Software Tools + GPUs have democratized the field
  - Diplomacy through back propagation!

- Whoo-hoo!
  - Anyone can get state of the art ASR with one machine and free software
  - Lowest barrier to entry in memory (recall Aurora 2)

- Uh-oh!
  - DNN systems achieve excellent performance without noise robustness
  - Aurora 4, voice search, Chime, AMI meetings

- What to make of all this? Is robustness dead?

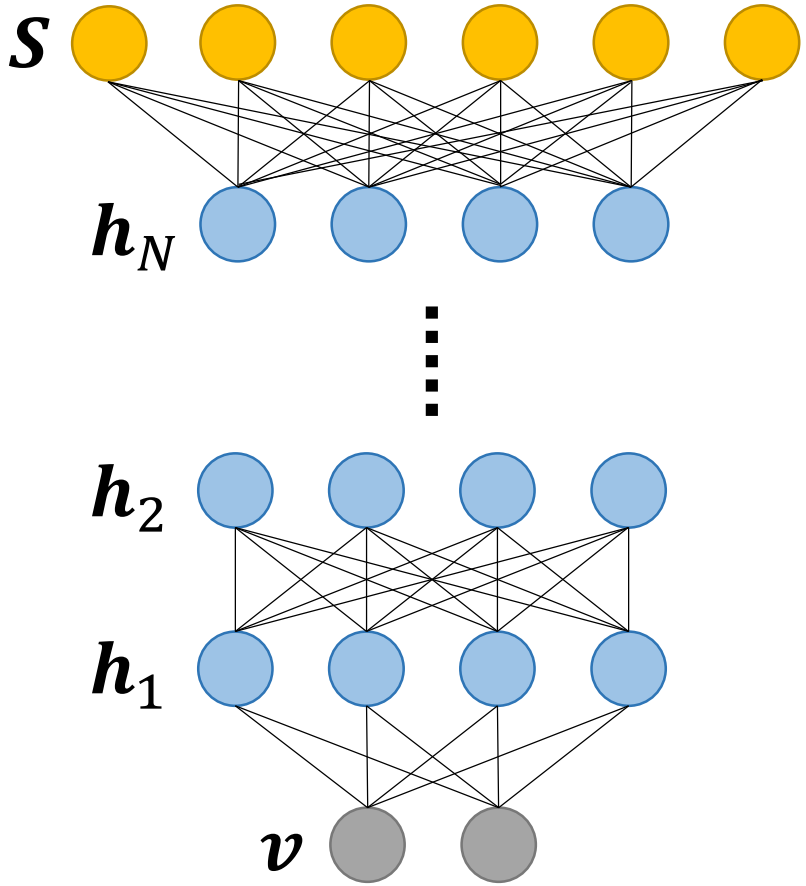
# This talk

- A look at DNN acoustic models with an emphasis on issues of robustness
  - What is a DNN?
  - Why do they work?
  - When do they fail?
  - How can they be improved?

- Goals:
  - Emphasize analysis over intensive comparisons and system descriptions
  - Open up the black box a bit
  - Show what DNNs do well so we can improve what they don’t

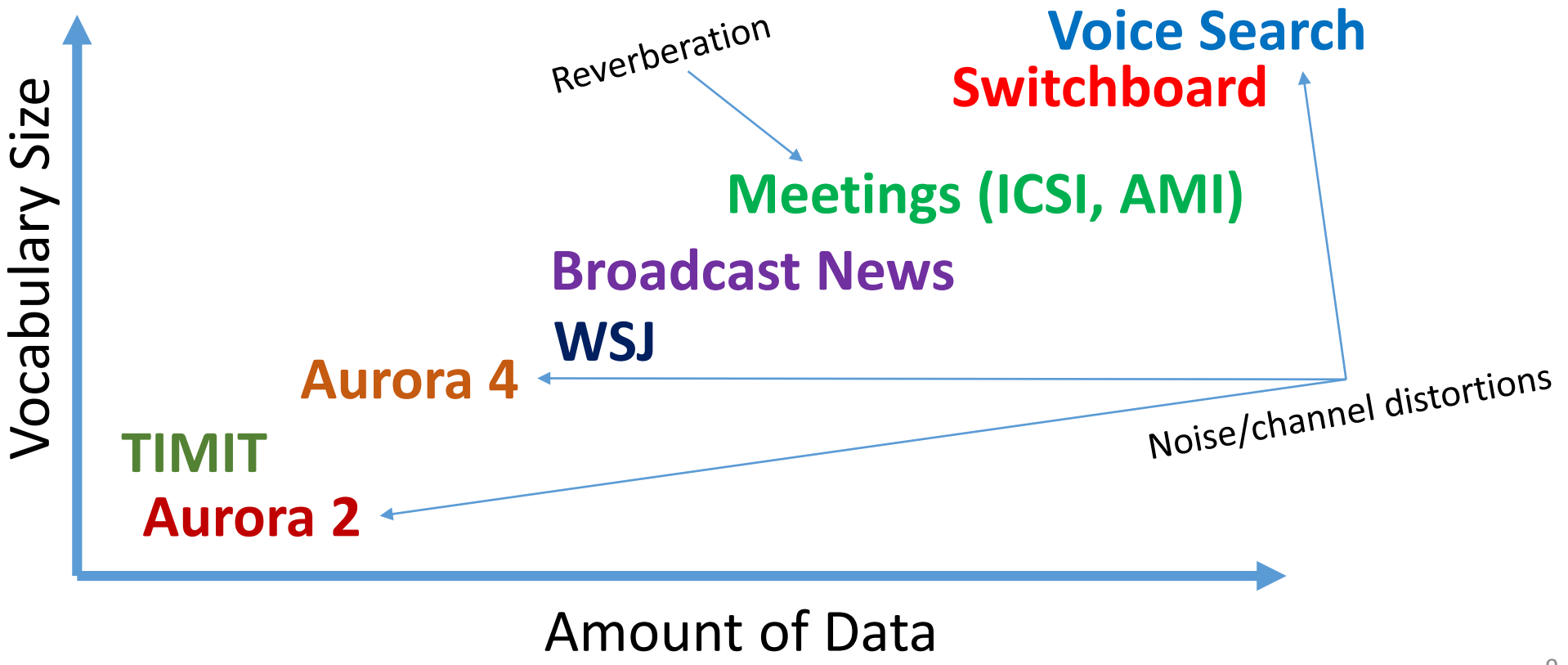
# A quick overview of deep neural networks

- Catchy name for MLP with “many” hidden layers
  - In: context window of frames
  - Out: senone posterior probability
- Training with back propagation to maximize the conditional likelihood at the frame or sequence level
- Optimization important & difficult, pre-training helps
- At runtime, convert posteriors to scaled likelihoods and decode as usual



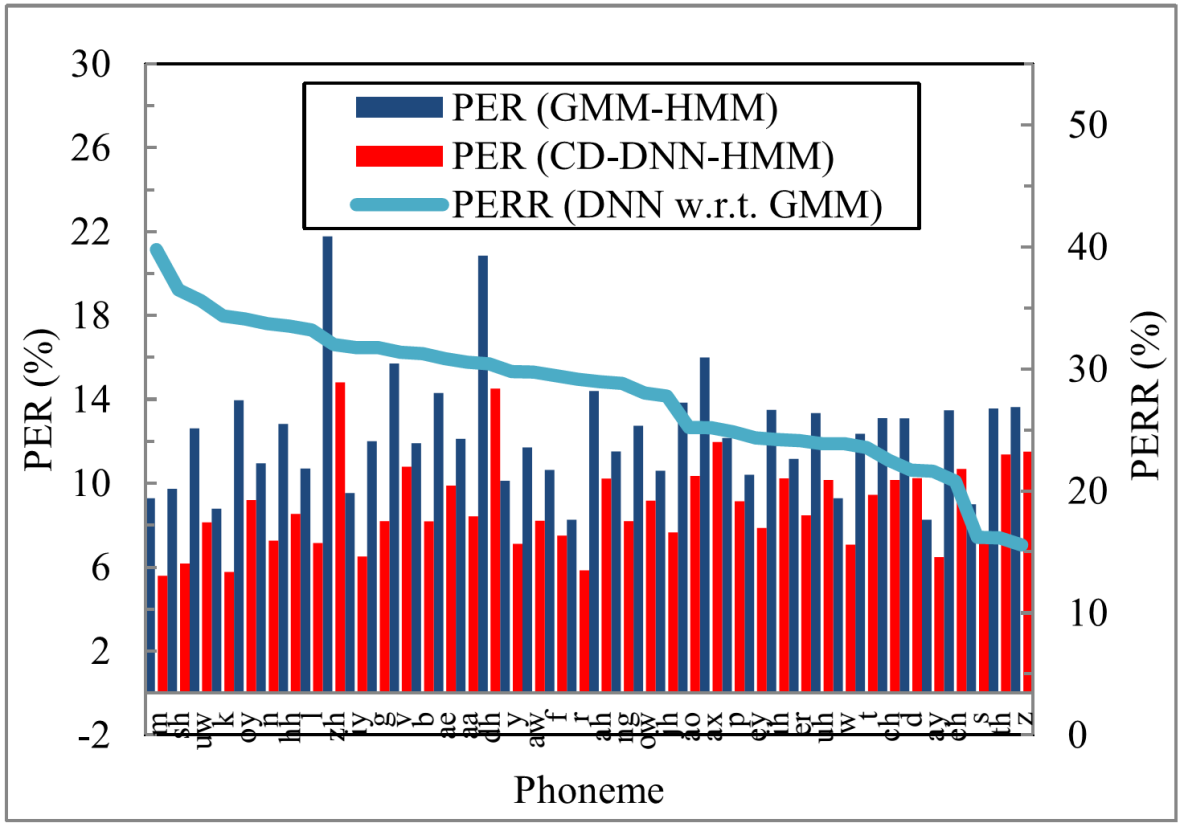
# Deep Neural Networks raise all boats…

- All tasks improve



# Deep Neural Networks raise all boats…

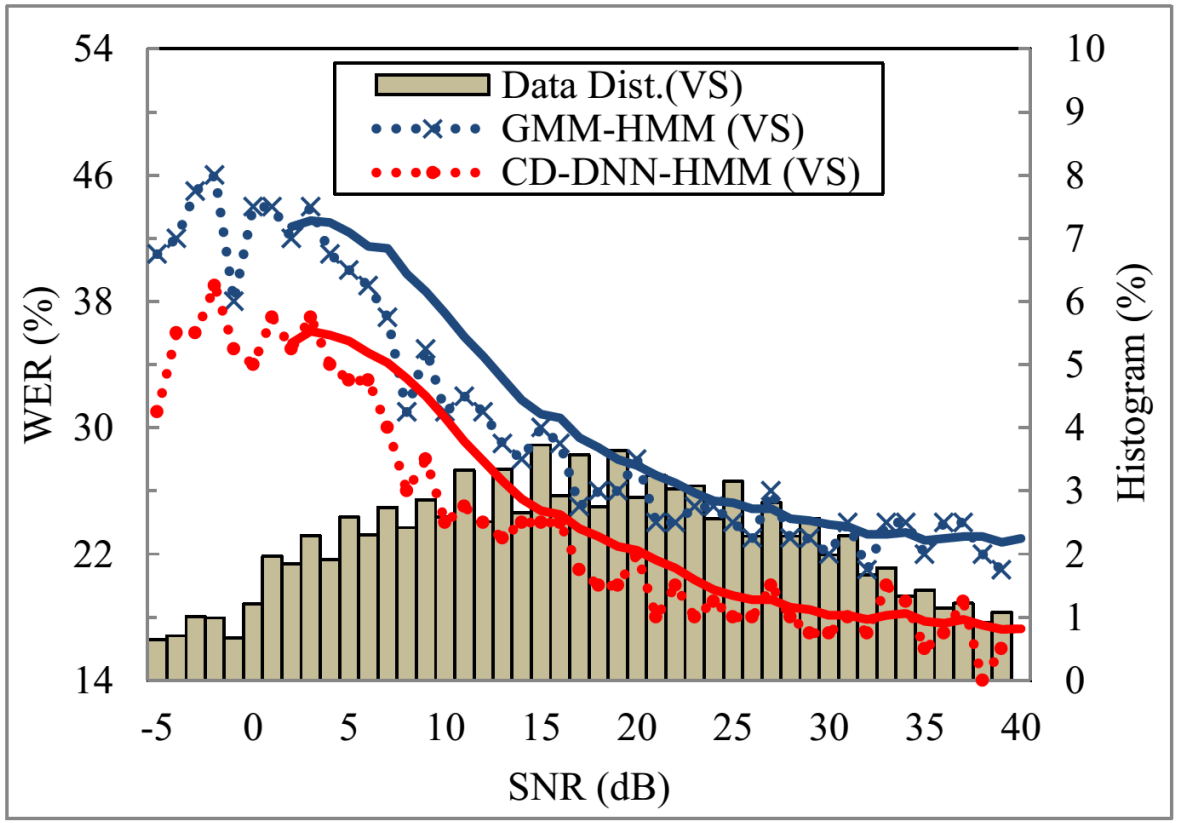
- All phonemes improve




[Huang 2014]

# Deep Neural Networks raise all boats…

- All SNRs improve




[Huang 2014]

# The power of depth

- Accuracy increases with depth

| # of Layers x # of Neurons | SWBD WER (%) [300hrs] | Aurora 4 WER(%) [10hrs] |
|---|---|---|
| 1 x 2k | 24.2 | --- |
| 3 x 2k | 18.4 | 14.2 |
| 5 x 2k | 17.2 | 13.8 |
| 7 x 2k | 17.1 | 13.7 |
| 9 x 2k | 17.0 | 13.9 |

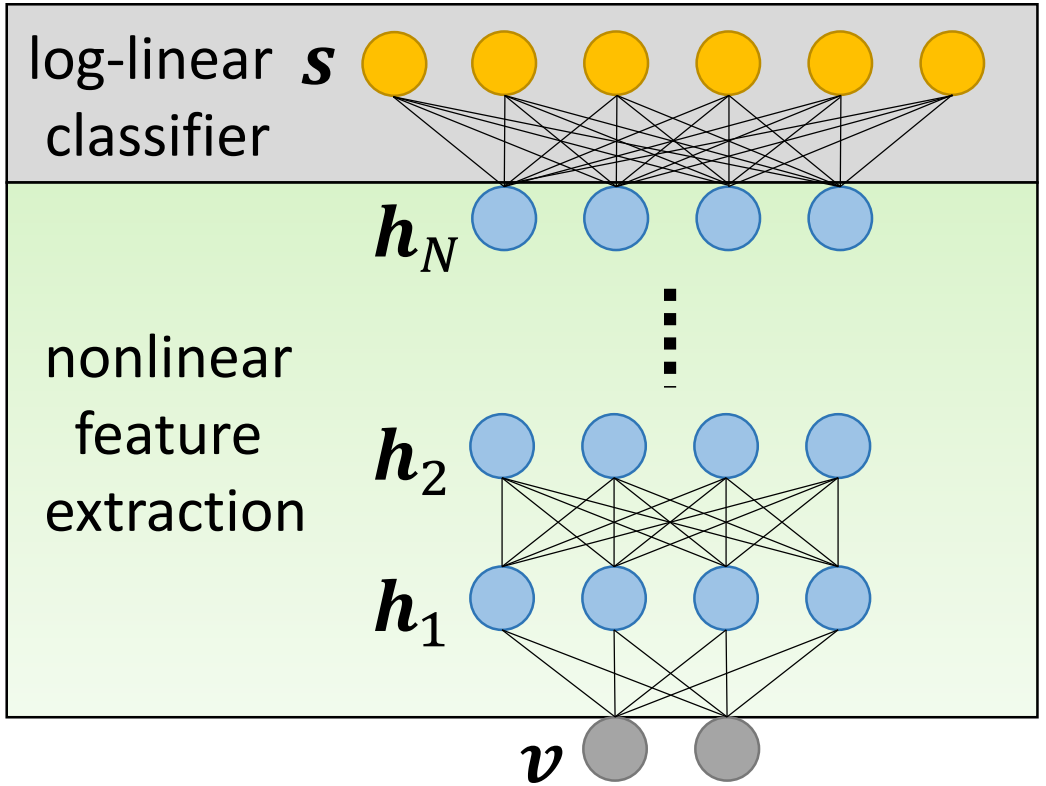
# The power of depth

- Depth is not just a way to add parameters

| # of Layers x # of Neurons | SWBD WER (%) [300hrs] | Aurora 4 WER(%) [10hrs] |
|---|---|---|
| 1 x 2k | 24.2 | --- |
| 3 x 2k | 18.4 | 14.2 |
| 5 x 2k | 17.2 | 13.8 |
| 7 x 2k | 17.1 | 13.7 |
| 9 x 2k | 17.0 | 13.9 |
| 1 x 16k | 22.1 | -- |

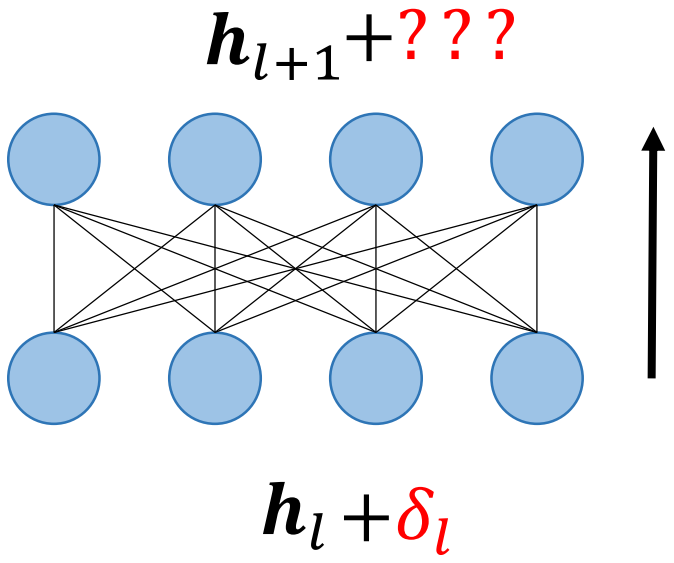
# Why have DNNs been successful?

- Many simple nonlinearities combine to form arbitrarily complex nonlinearities
- Single classifier shares all parameters and internal representations
- Joint feature learning & classifier design
  - Unlike tandem or bottleneck systems
- Features at higher layers more invariant and discriminative than at lower layers



# How is invariance achieved?

- How do DNNs achieve invariance in the representation?
- Consider forward propagation: $\boldsymbol{h}_{l+1} = \sigma(\boldsymbol{W}_l \boldsymbol{h}_l) = f(\boldsymbol{h}_l)$



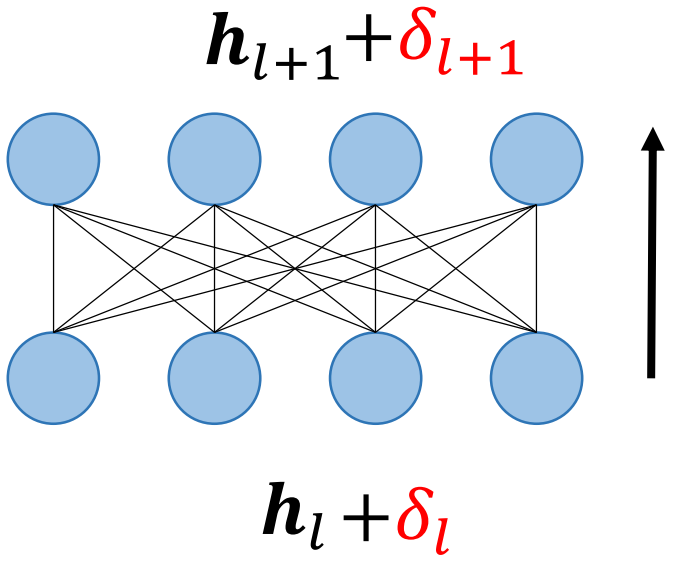
# How is invariance achieved?

- Forward propagation: $h_{l+1} = \sigma(W_l h_l) = f(h_l)$

$$\delta_{l+1} = \sigma\big(W_l(h_l + \delta_l)\big) - \sigma(W_l h_l)$$

$$\delta_{l+1} \approx \sigma'(W_l h_l) W_l^T \delta_l$$

$$\frac{\partial f}{\partial h} \approx \frac{f(h+\delta) - f(h)}{\delta}$$

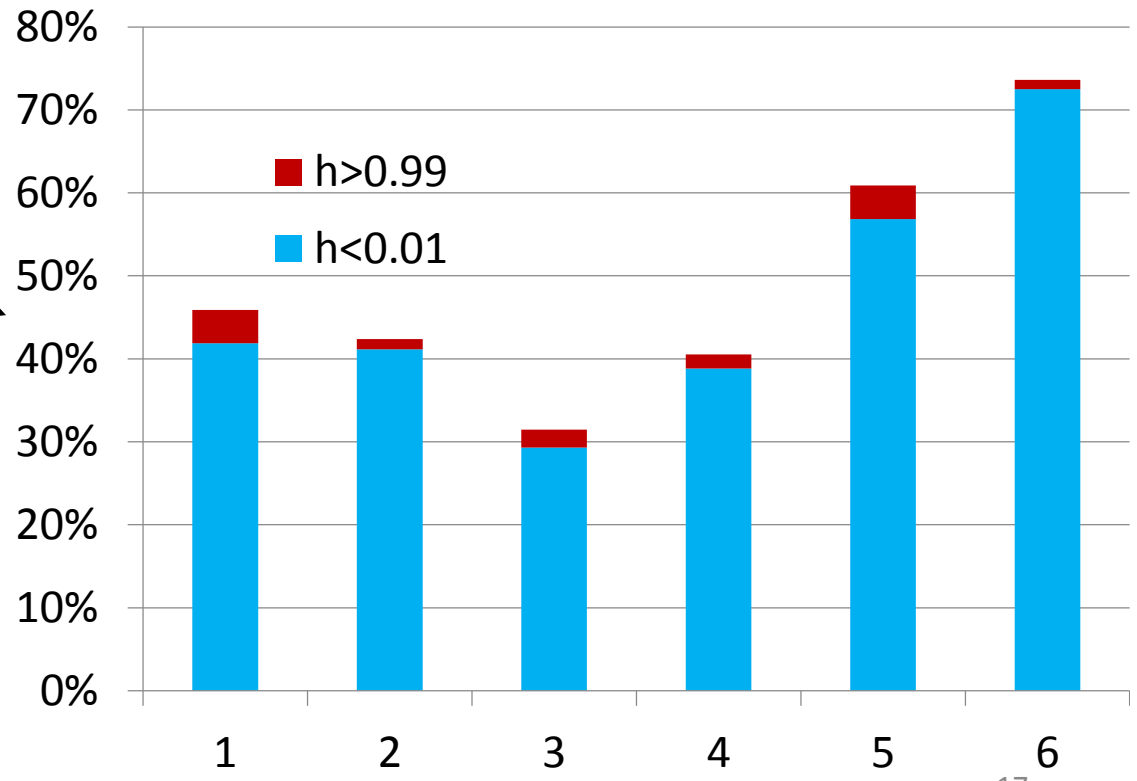



$$\|\delta_{l+1}\| < \left\|diag\big(h_{l+1} \circ (1 - h_{l+1})\big) W_l^T\right\| \|\delta_l\|$$

# How is invariance achieved?

$$\|\delta_{l+1}\| < \left\|diag\left(h_{l+1} \circ (1 - h_{l+1})\right) W_l^T\right\| \|\delta_l\|$$

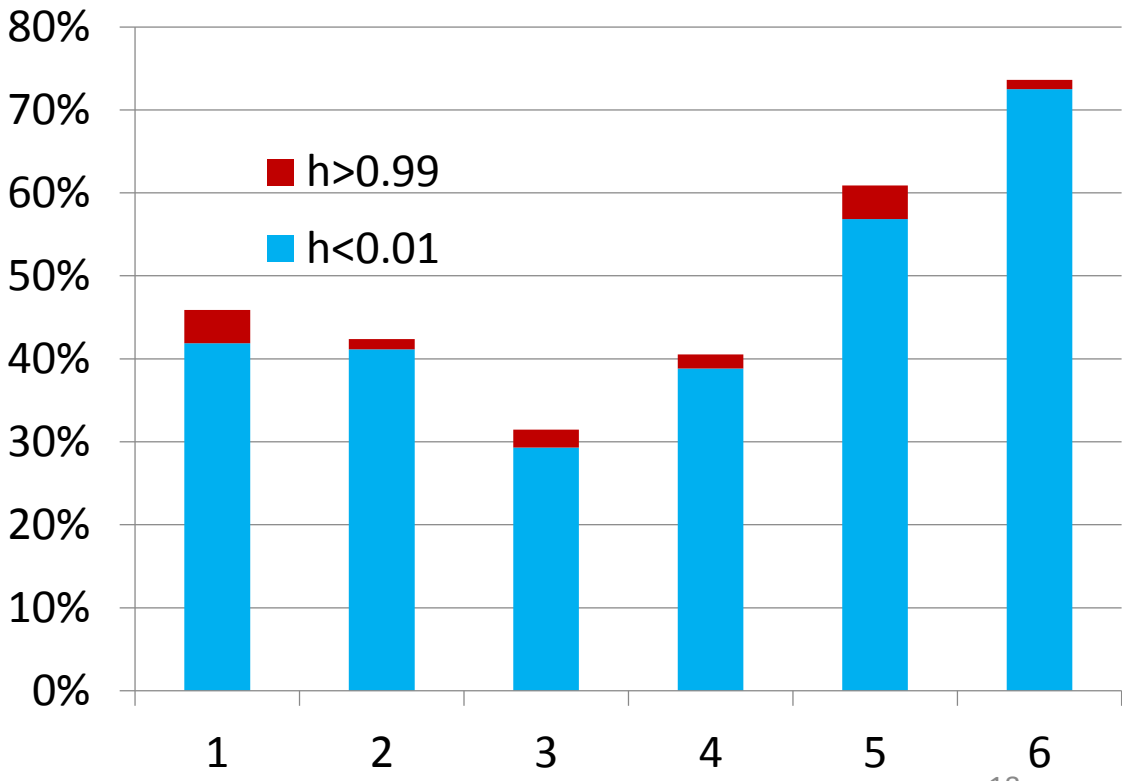
- The first term always <=0.25
- Much smaller when saturated
  - Higher layers are more sparse



# How is invariance achieved?

$$\|\delta_{l+1}\| < \left\|diag\left(h_{l+1} \circ (1 - h_{l+1})\right) W_l^T\right\| \|\delta_l\|$$

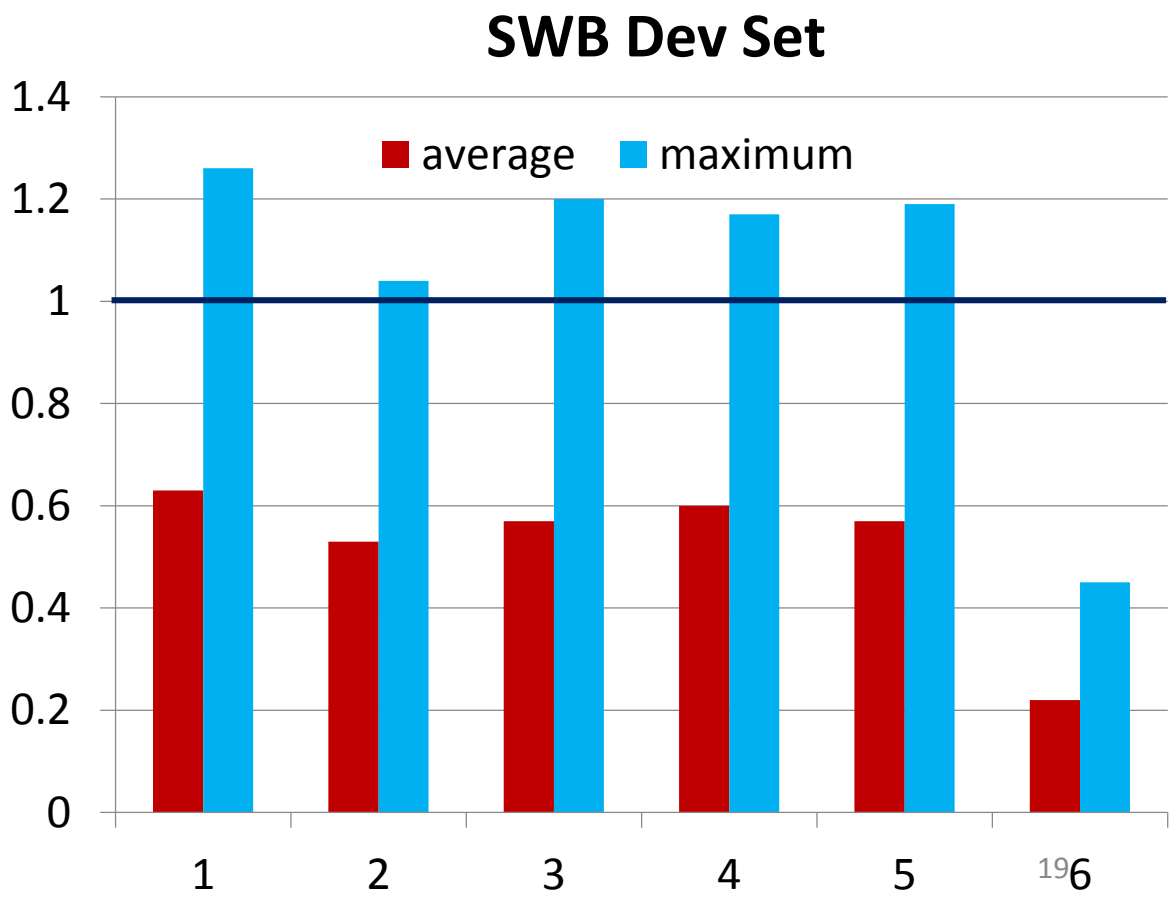
- The first term always <=0.25
- Much smaller when saturated
  - Higher layers are more sparse
- For large networks, most weights are very small
  - SWBD: 98% of weights < 0.5



# How is invariance achieved?

- "$\delta$ gain" < 1 on average
- Variation shrinks from one layer to the next
- Maximum is > 1
  - Enlarge $\delta$ near decision boundaries
  - More discriminative
- For input "close" to training data, each layer improves invariance
  - Increases robustness

$$\left\| diag\left(h_{l+1} \circ (1 - h_{l+1})\right) W_l^T \right\|$$



# Visualizing invariance with t-SNE

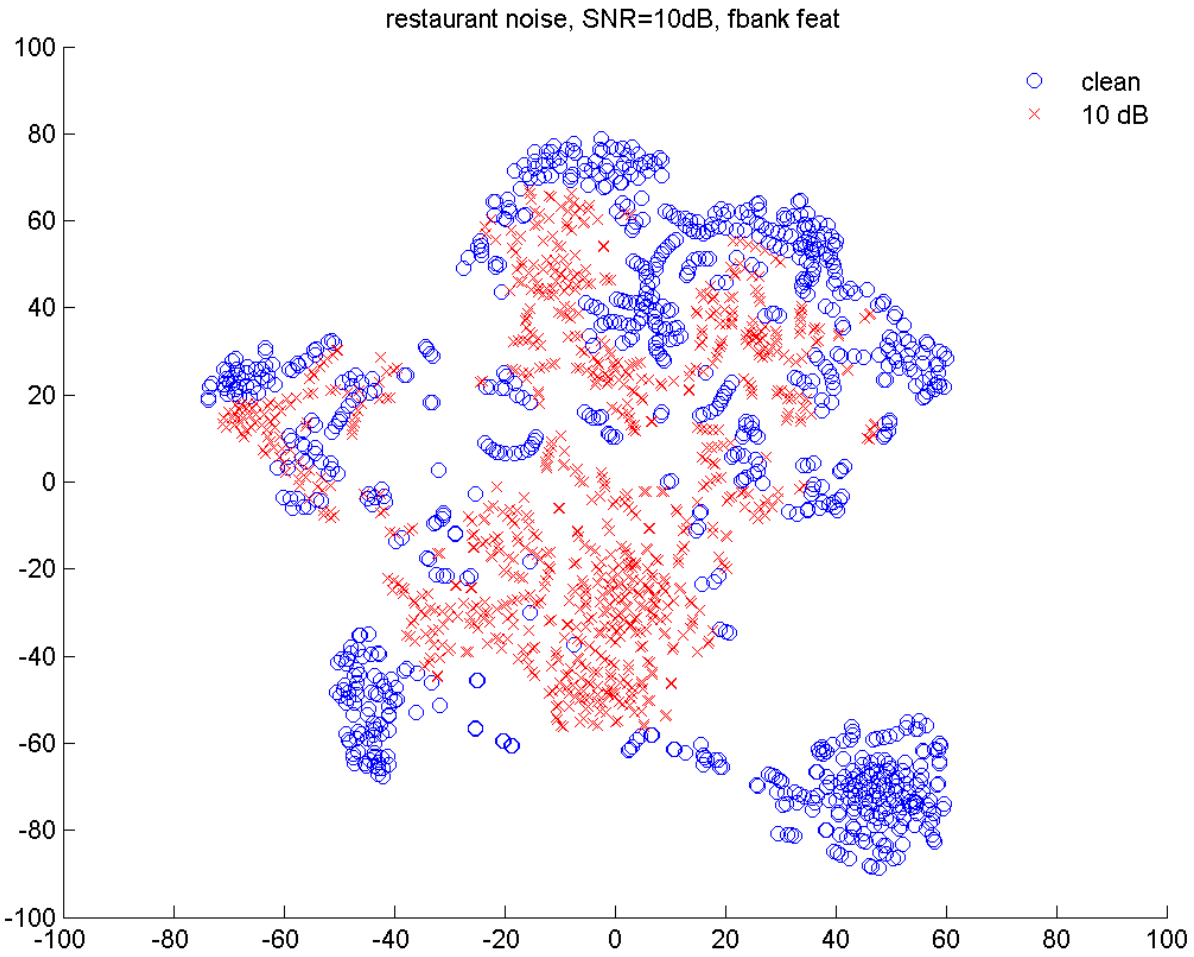
[van der Maaten 2008]

- A way to visualize high dimensional data in a low dimensional space
- Preserves neighbor relations
- Use to examine input and internal representations of a DNN

- Consider a parallel pair of utterances:
  - a noise-free utterance recorded with a close-talking microphone
  - the same utterance corrupted by restaurant noise at 10 dB SNR

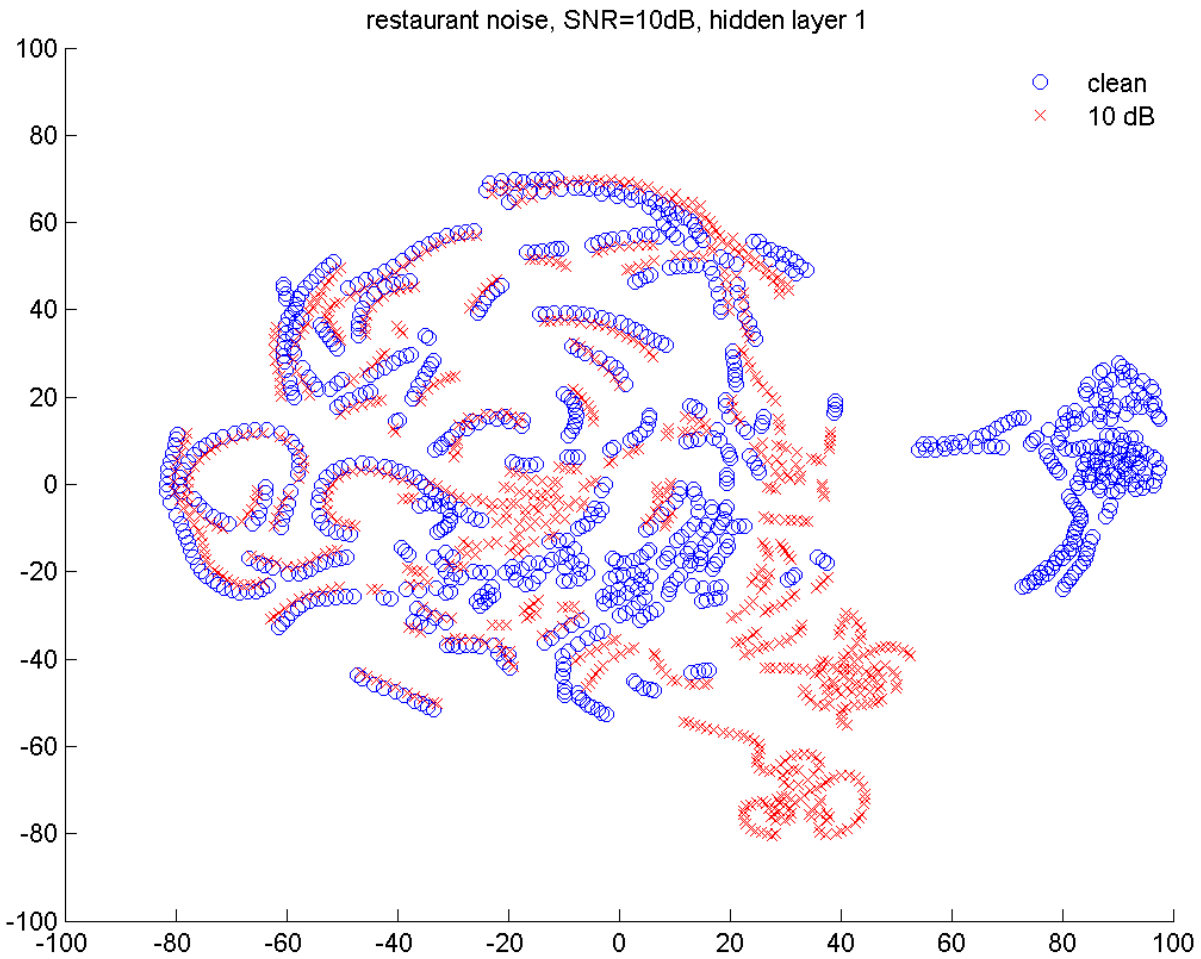
# Visualizing invariance with t-SNE

- Features



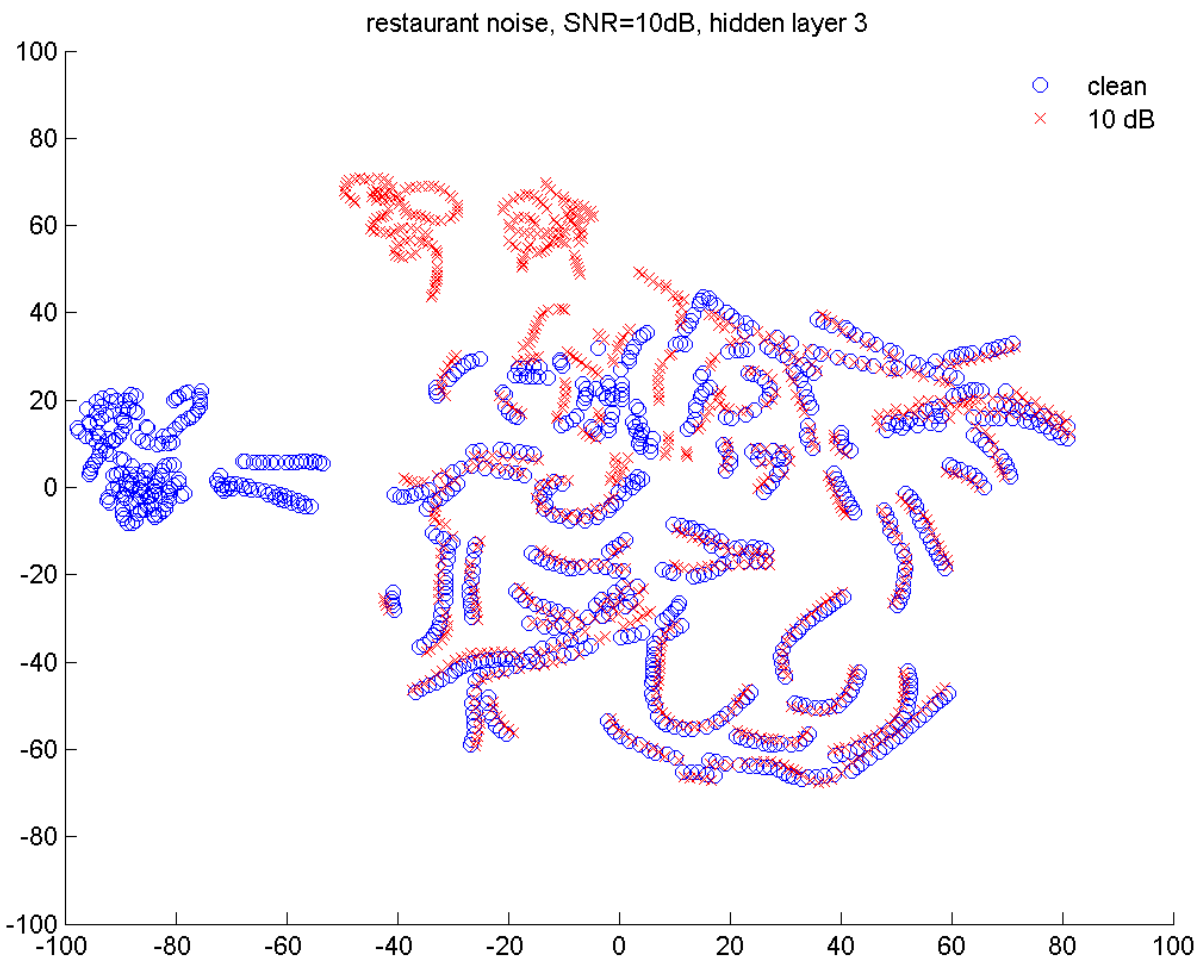
# Visualizing invariance with t-SNE

- 1st layer



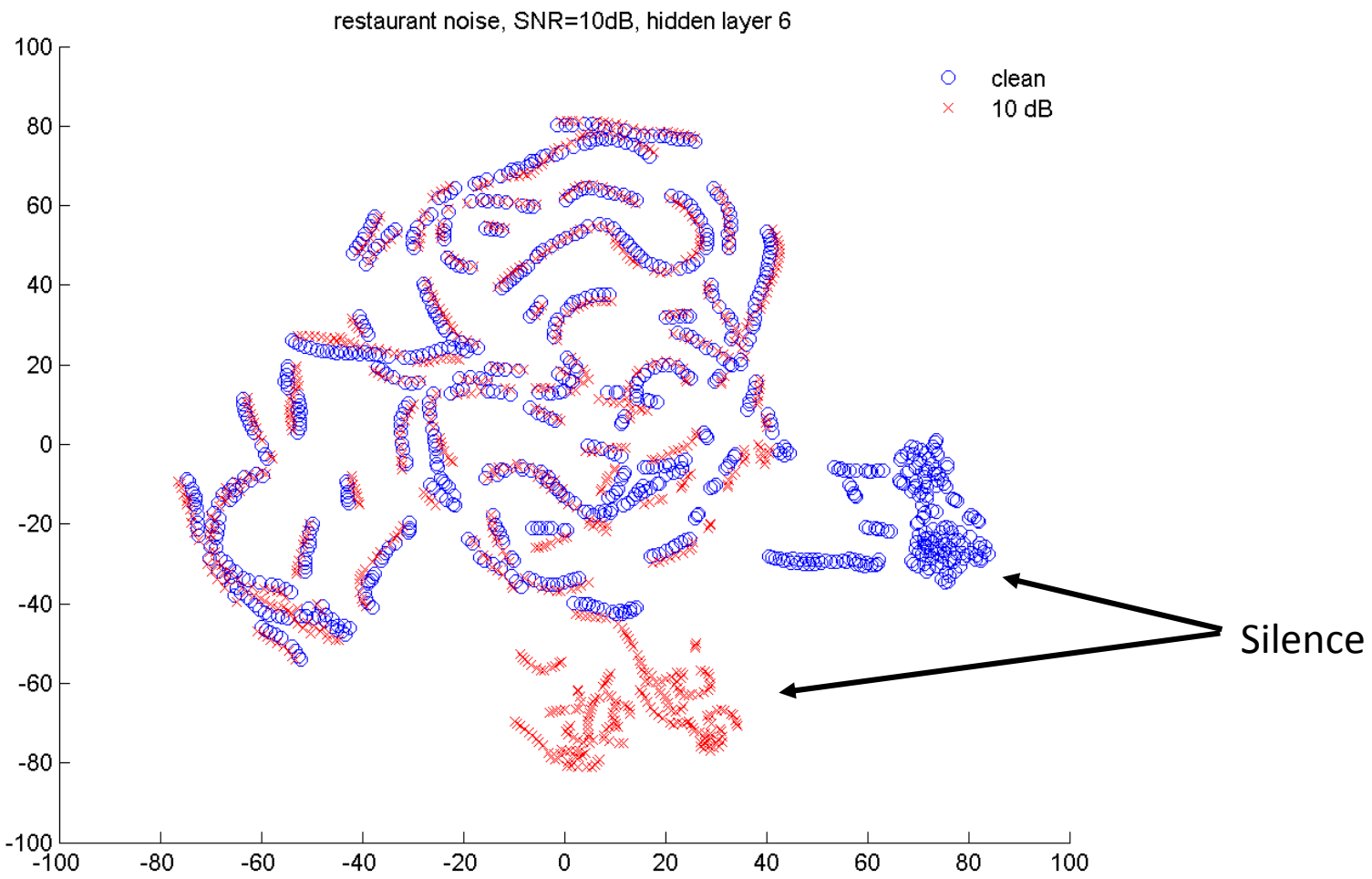
# Visualizing invariance with t-SNE

- 3rd layer



# Visualizing invariance with t-SNE

- 6th layer



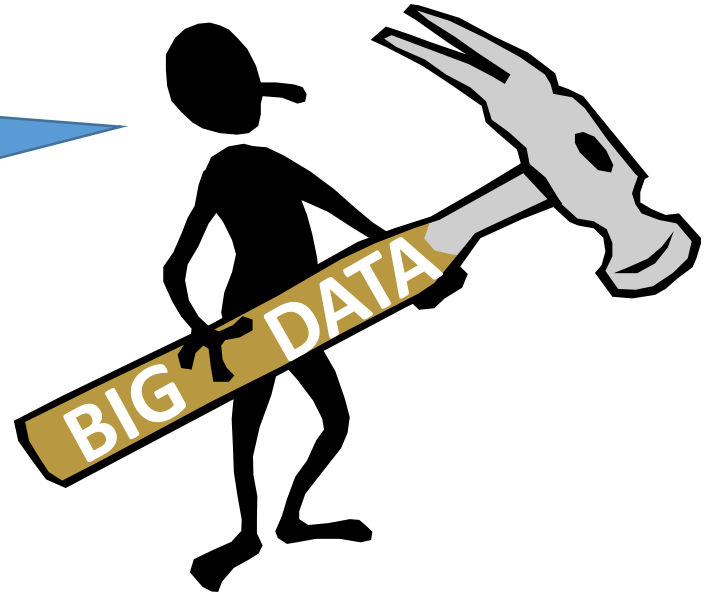
# Invariance improves robustness

- DNNS are robust to small variations of the training data
- Explicitly normalizing these variations is less important/effective
  - Network is already doing it
- Removing too much variability from the data may hinder generalization

| Preprocessing Technique | Task | DNN Relative Imp |
|---|---|---|
| VTLN (speaker) | SWBD | <1% [Seide 2011] |
| C-MMSE (noise) | Aurora4/VS | <0% [Seltzer 2013] |
| IBM/IRM Masking (noise) | Aurora 4 | <0% [Sim 2014] |

# The end of robustness?

"The more training data used, the greater the chance that a new sample can be trivially related to samples in the training data, thereby lessening the need for any complex reasoning that may be beneficial in the cases of sparse training data." [Brill 2002]




- *"The unreasonable effectiveness of data"* [Halevy 2009]
- In DNN terms: with more data, the likelier a new sample lies within $\delta$ of a training example

# (Un)fortunately, data is not a panacea

- Even in the best cases, performance gaps persist
  - Noisy is 2X WER of Clean (Aurora 4, VS)
  - Unseen environments 2X WER of seen noises with MST (Aurora 4)
  - Farfield is 2X WER of Close-talk (Meetings)
- Some scenarios cannot support large training sets
  - Low resource languages
- Mismatch is sometimes unavoidable
  - New devices, environments
- Sometimes modeling assumptions are wrong
  - Speech separation, reverberation

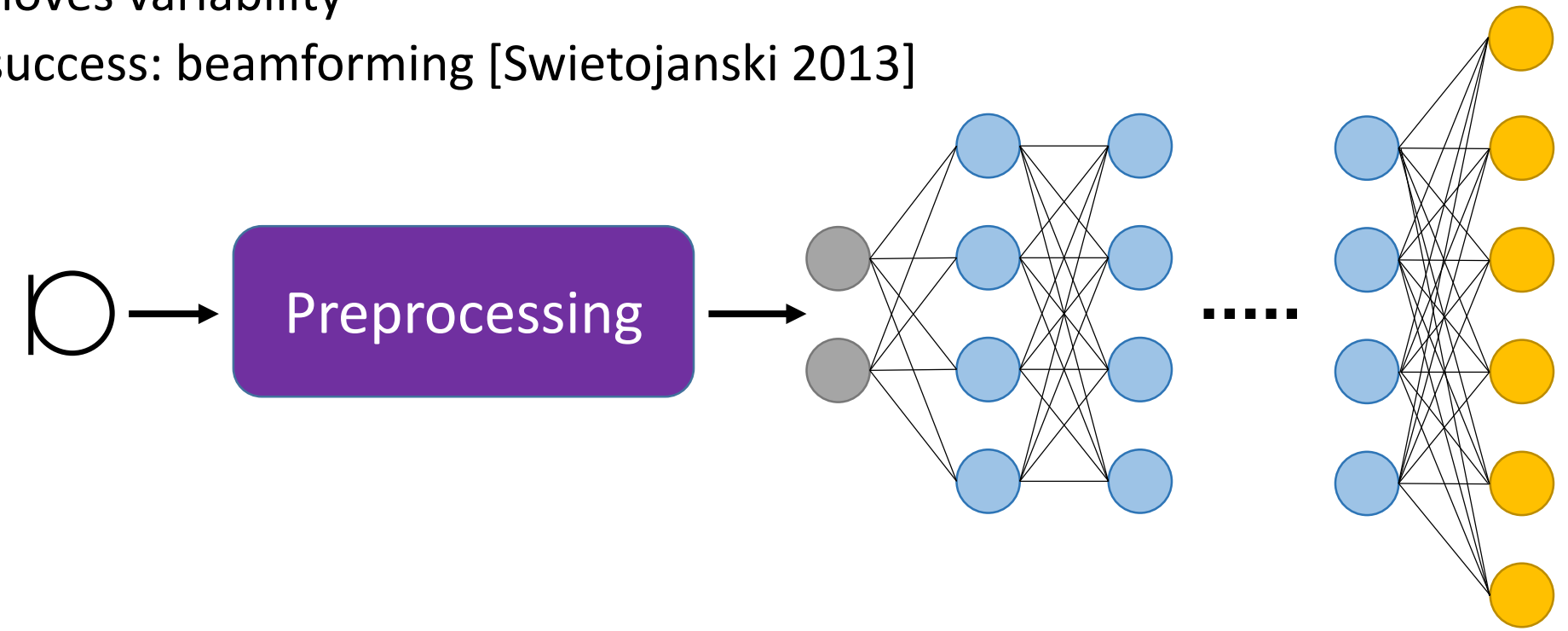
# Robustness: the triumphant return!

- Systems still need to be more robust to variability
  - speaker, environment, device

- Guiding principles:
  - Exposure to variability is good (multi-condition training)
  - Limiting variability can harm performance
  - Close relationship to desired objective function is desirable

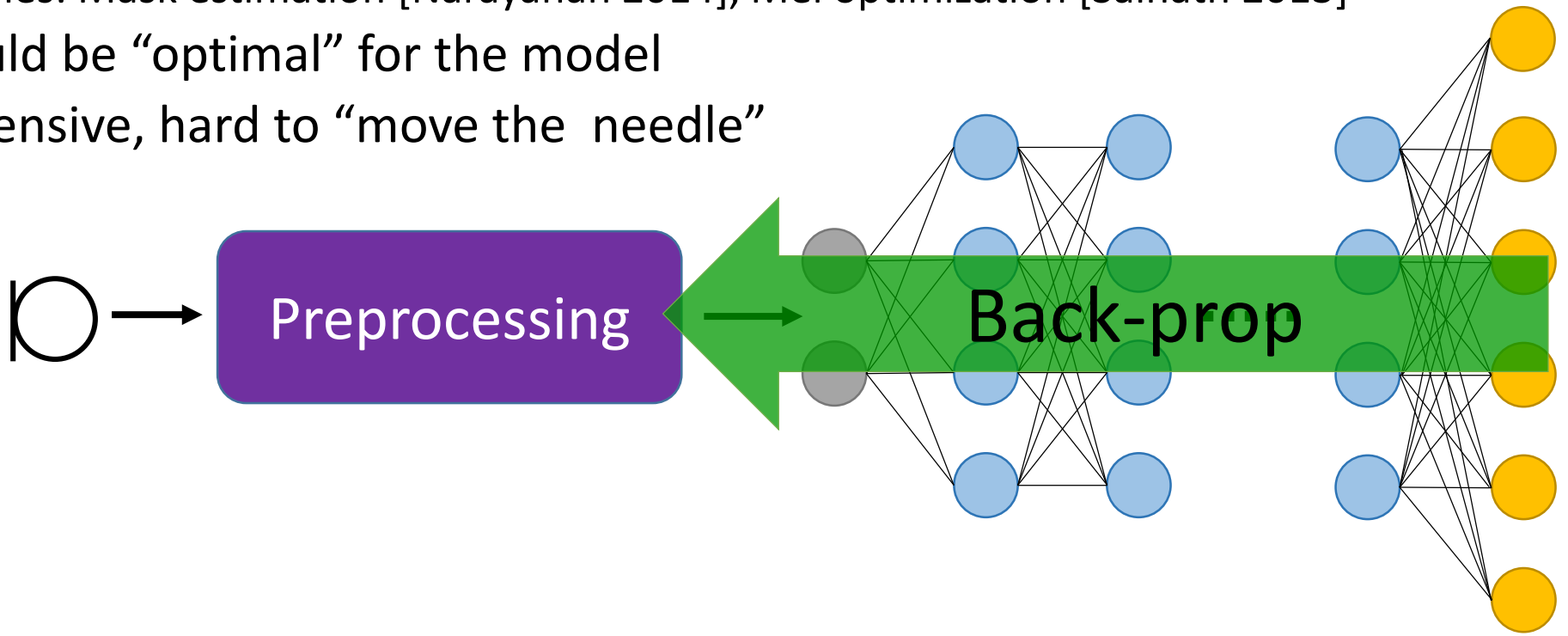
# Approach 1: Decoupled Preprocessing

- Processing independent of downstream activity
  - Pro: simple
  - Con: removes variability
  - Biggest success: beamforming [Swietojanski 2013]



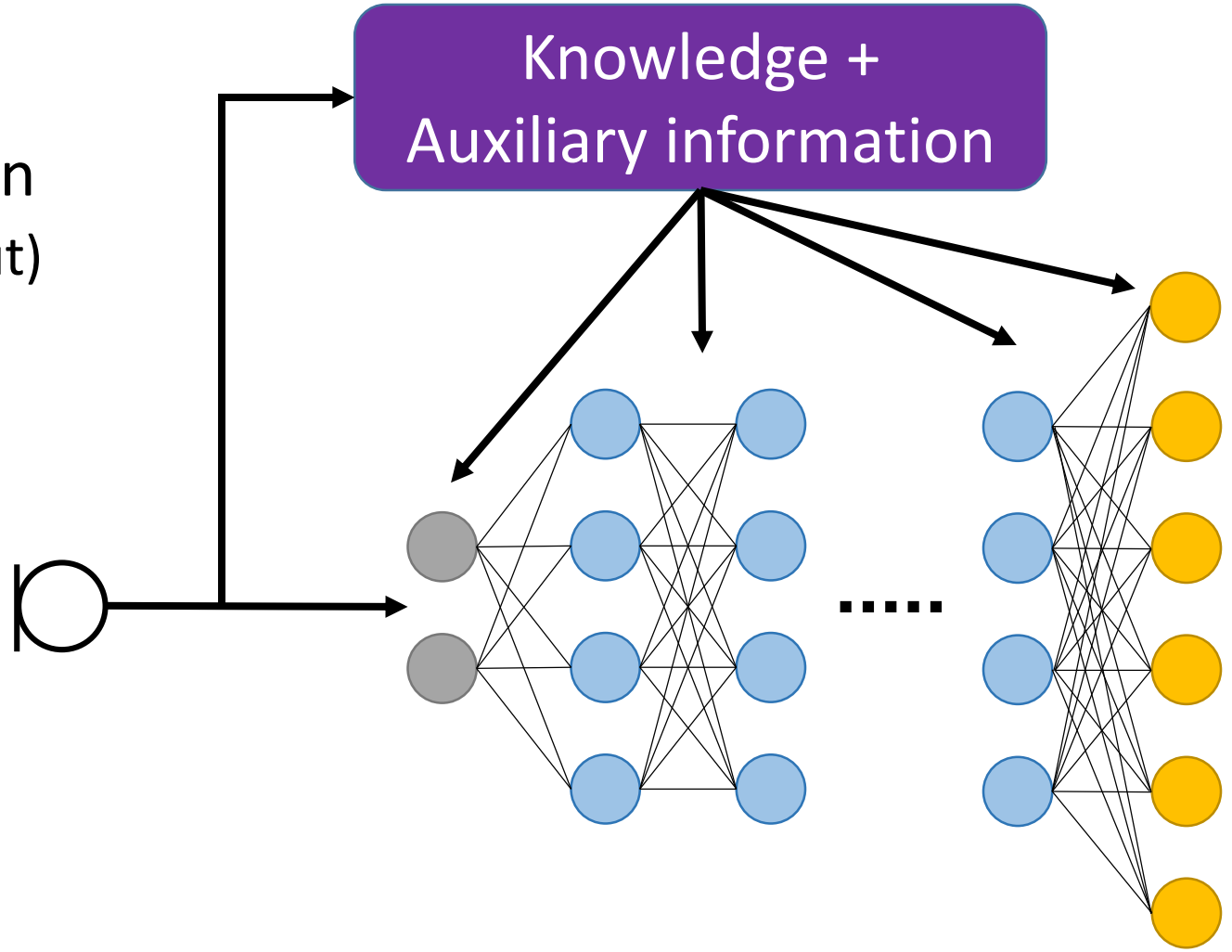
# Approach 2: Integrated Preprocessing

- Treat preprocessing as initial “layers” of model
  - Optimize parameters with back propagation
    - Examples: Mask estimation [Narayanan 2014], Mel optimization [Sainath 2013]
  - Pro: should be “optimal” for the model
  - Con: expensive, hard to “move the  needle”



# Approach 3: Augmented information

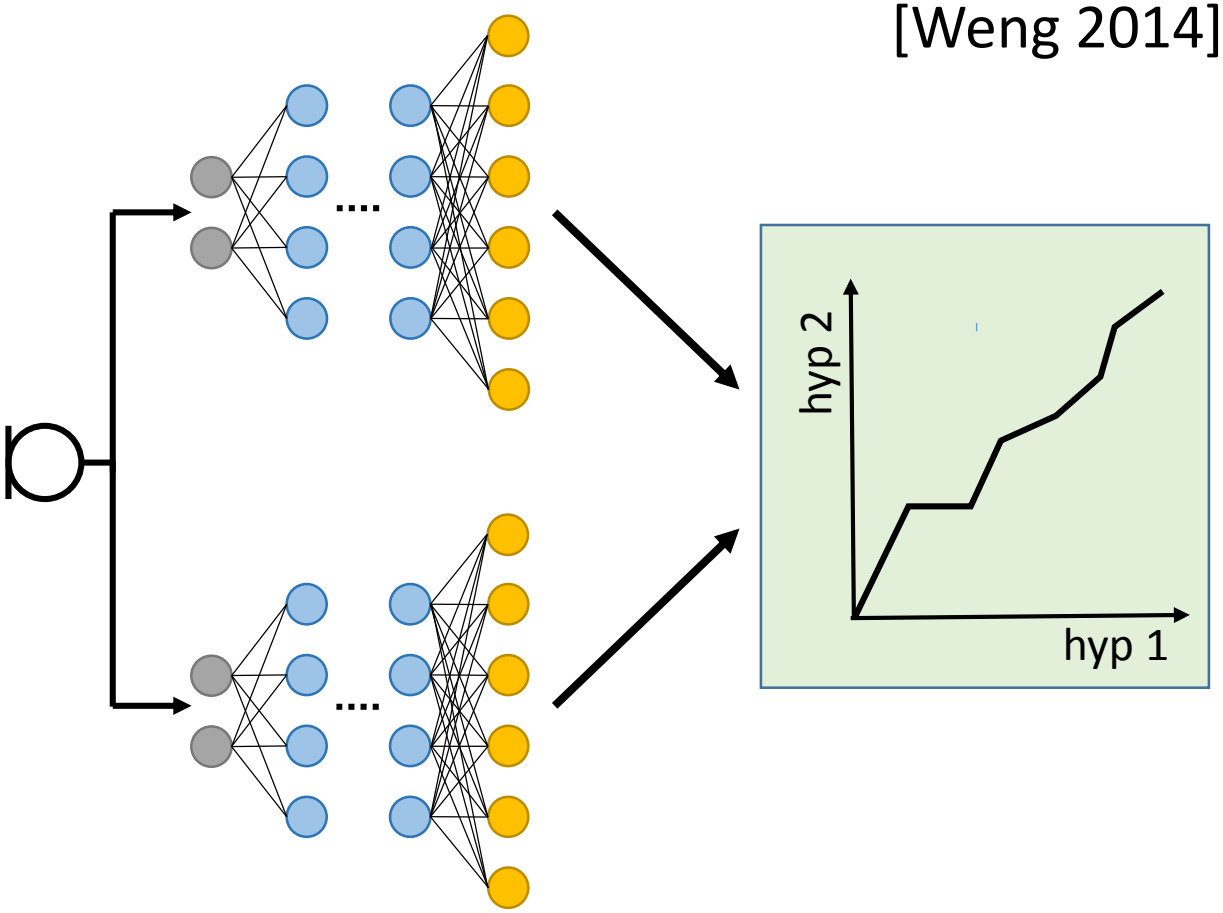
- Augment model with informative side information
  - Nodes (input, hidden, output)
  - Objective function
- Pros:
  - preserves variability
  - adds knowledge
  - operates on representation
- Con:
  - No physical model



# Example: co-channel speech separation

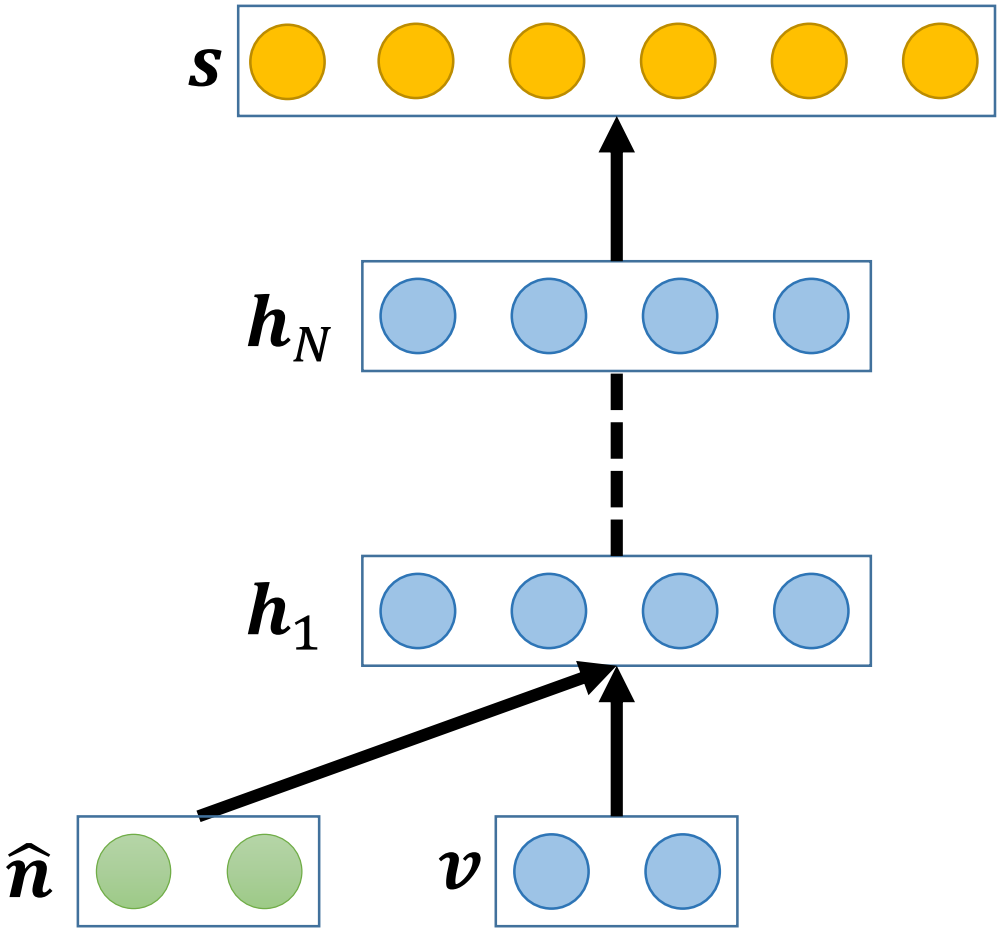
- Create multi-style training data
- Train 2 DNNs
  - Frame-level SNR to label
- Jointly decode both hypotheses
  - Add trained adaptive penalty to penalize frequent switching
- Speech Separation Challenge:
  - IBM Superhuman: 21.6% WER
  - Proposed: 20.0% WER

[Weng 2014]



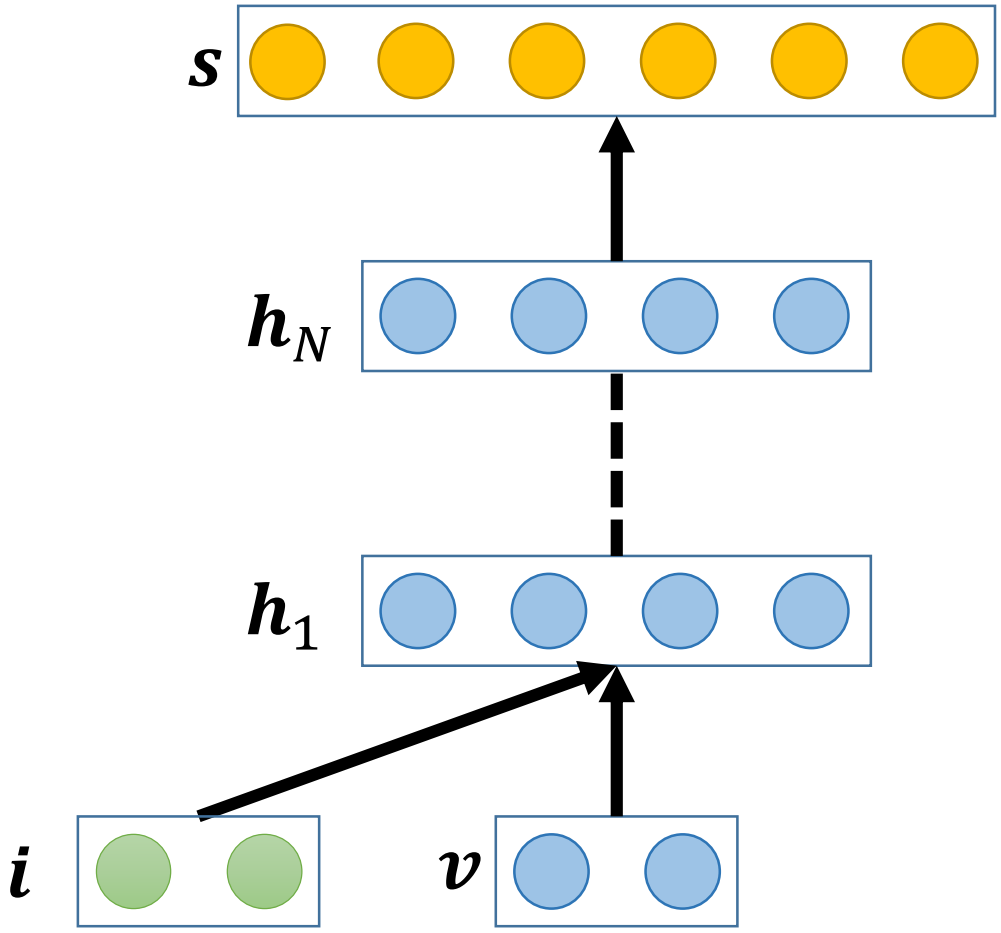
# Example 2: noise aware training/adaptation

- Similar motivation to noise-adaptive training of GMM acoustic models
- Give network cues about source of variability [Seltzer 2013]
- Preserve variability in training data



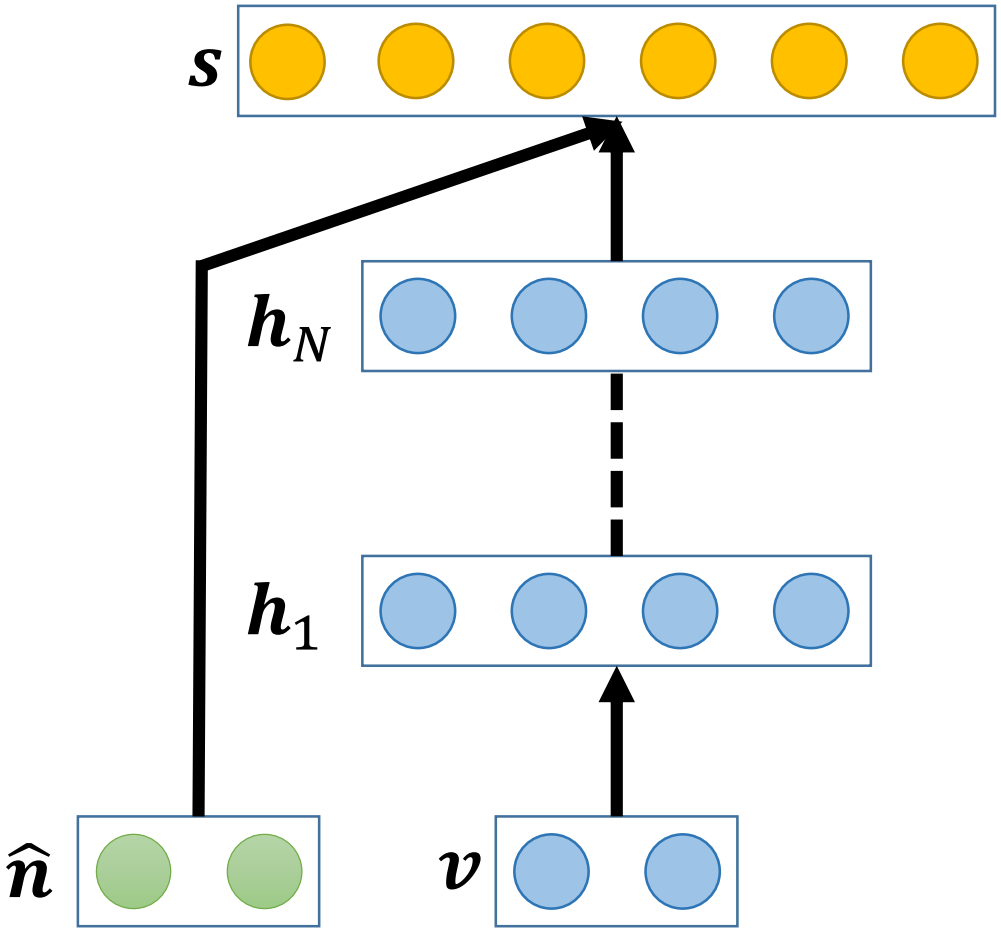
# Example 2: noise aware training/adaptation

- Similar motivation to noise-adaptive training of GMM acoustic models
- Give network cues about source of variability [Seltzer 2013]
- Preserve variability in training data
- Works for speaker adaptation [Saon 2013]



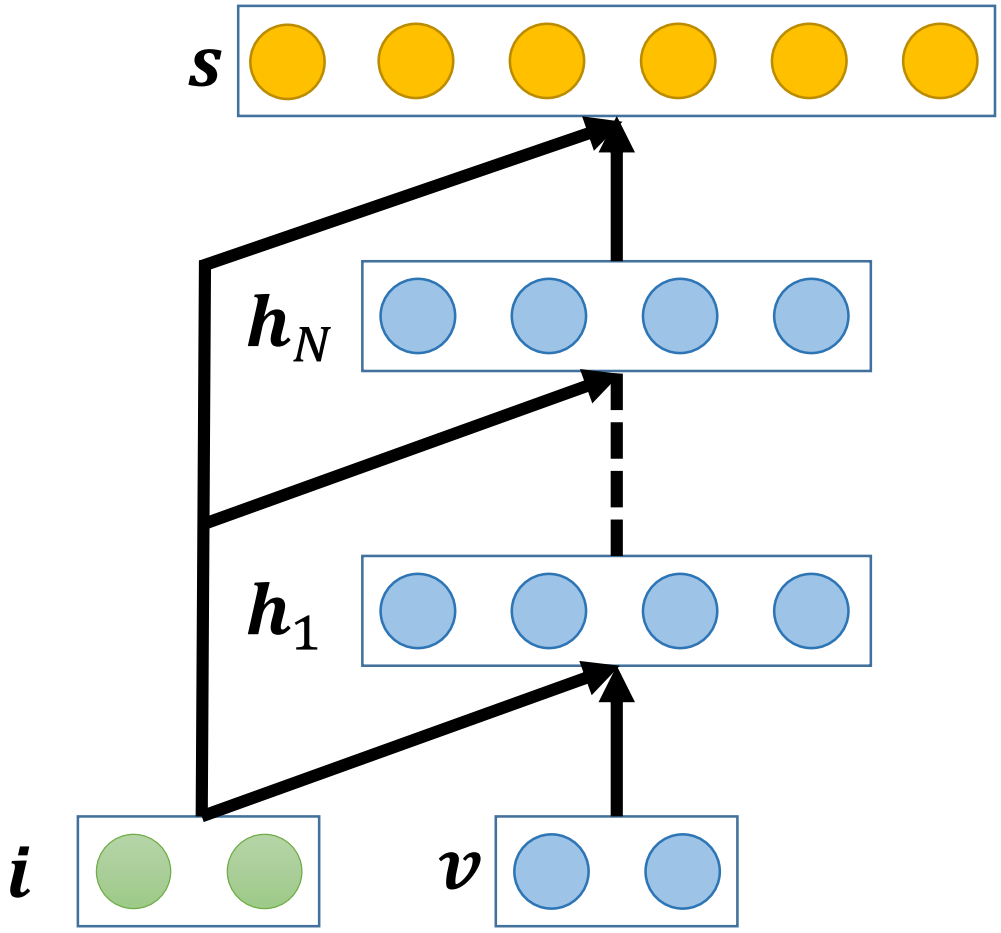
# Example 2: noise aware training/adaptation

- Similar motivation to noise-adaptive training of GMM acoustic models
- Give network cues about source of variability [Seltzer 2013]
- Preserve variability in training data
- Works for speaker adaptation [Saon 2013]
- …and noise adaptation [Li 2014]



# Example 2: noise aware training/adaptation

- Similar motivation to noise-adaptive training of GMM acoustic models
- Give network cues about source of variability [Seltzer 2013]
- Preserve variability in training data
- Works for speaker adaptation [Saon 2013]
- …and noise adaptation [Li 2014]
- …at all layers [Xue 2014]



# Summary

- DNNs have had a dramatic impact on speech recognition
- DNNs are incredibly robust to unwanted variability including noise
- Robustness is achieved through feature invariance
- Invariance is achieved through the combination of large training sets and deep networks
- Several areas where performance still suffers and there are opportunities for improvement
- (At least) three architectures for incorporating robustness into DNNs
- It’s still early days…lots of exciting work to do!

# Conclusion

- Is robustness dead?

> *The reports of my death have been greatly exaggerated.* -M. Twain

- Long live robustness!

# Thank you!

# References


- D. Yu, M. L. Seltzer, J. Li, J.-T. Huang, F. Seide, "Feature learning in deep neural networks – studies on speech recognition tasks," in *Proc. ICLR*, 2013
- F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. ASRU*, 2011
- Y. Huang, D. Yu, C. Liu, and Y. Gong, "A comparative analytic study on the Gaussian mixture and context-dependent deep neural network hidden markov models," *in submission*
- M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of noise robustness of deep neural networks," in *Proc. ICASSP*, 2013
- L. J. P. van der Maaten and G.E. Hinton, "Visualizing high-dimensional data using t-SNE," Journal of Machine Learning Research 9(Nov):2579-2605, 2008
- C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Single-channel mixed speech recognition using deep neural networks," in *Proc. ICASSP*, 2014
- J. Li, J.-T. Huang, and Y. Gong, "Factorized adaptation for deep neural network," in *Proc. ICASSP*, 2014
- E. Brill, J. Lin, M. Banko, S. Dumais and A. Ng, "Data-intensive question answering," in *Proc. TREC*, 2001
- A. Halevy, P. Norvig, F. Pereira, "The unreasonable effectiveness of data," *Intelligent Systems, IEEE* , vol.24, no.2, pp.8-12, Mar-Apr 2009
- A. Narayanan and D. Wang, "Joint noise adaptive training for robust automatic speech recognition," in *Proc. ICASSP*, 2014
- T. N. Sainath, B. Kingsbury, A. Mohamed, and B. Ramabhadran, "Learning filter banks within a deep neural network framework," in *Proc. ASRU*, 2013.
- B. Li and K. C. Sim, "An ideal hidden-activation mask for deep neural networks based noise-robust speech recognition," in *Proc. ICASSP*, 2014
- S. Xue, O. Abdel-Hamid, H. Jiang, and L. Dai, "Direct adaptation of hybrid DNN/HMM model for fast speaker adaptation in LVCSR based on speaker code," in *Proc. ICASSP*, 2014
- G. Saon, H. Soltau, M. Picheny, and D. Nahamoo, "Speaker adaptation of neural network acoustic models using i-vectors," in *Proc. ASRU*, 2013.
- P Swietojanski, A Ghoshal, and S Renals, "Hybrid acoustic models for distant and multichannel large vocabulary speech recognition," in *Proc. ASRU*, 2013