Deep Learning for Speech and Language

Yoshua Bengio, U. Montreal

NIPS'2009 Workshop on Deep Learning for Speech Recognition and Related Applications

December 12, 2009

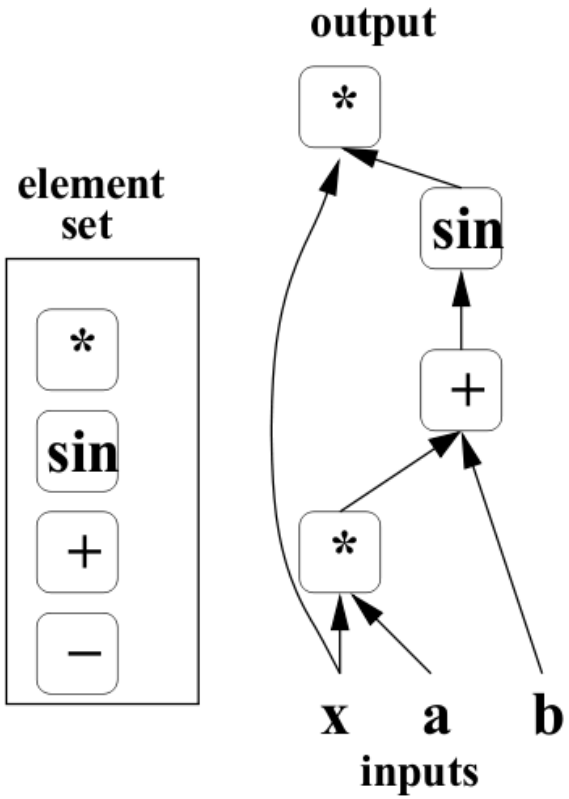
Interesting Experimental Results with Deep Architectures

- Beating shallow neural networks on vision and NLP tasks
- Beating SVMs on visions tasks from pixels (and handling dataset sizes that SVMs cannot handle in NLP)
- Reaching or beating state-of-the-art performance in NLP and phoneme classification
- Beating deep neural nets without unsupervised component
- Learn visual features similar to V1 and V2 neurons as well as auditory cortex neurons

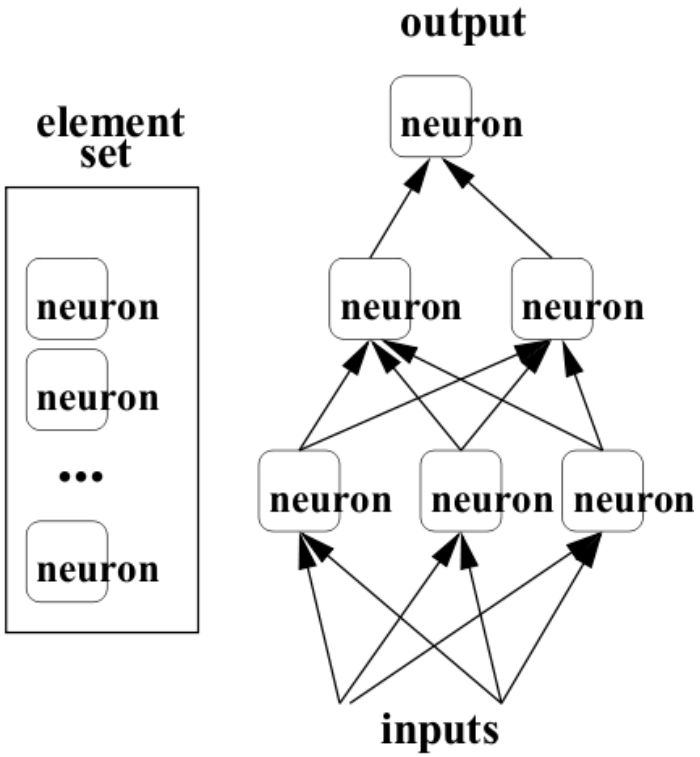
Deep Motivations

- Brains have a deep architecture
- Humans organize their ideas hierarchically, through composition of simpler ideas
- Unsufficiently deep architectures can be exponentially inefficient
- Distributed (possibly sparse) representations are necessary to achieve non-local generalization
- Multiple levels of latent variables allow combinatorial sharing of statistical strength

Architecture Depth



Depth = 4

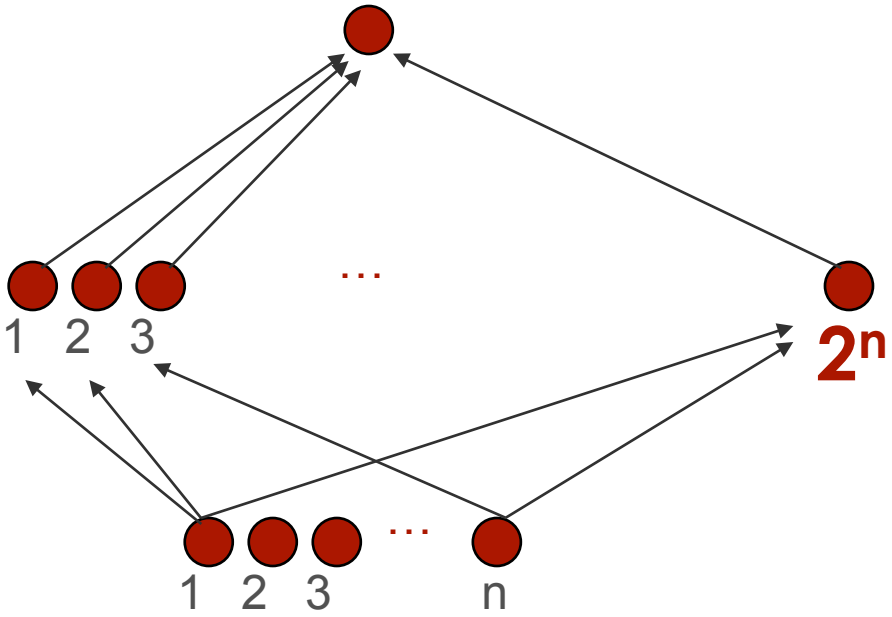


Depth = 3

Deep Architectures are More Expressive

Theoretical arguments:

2 layers of { Logic gates / Formal neurons / RBF units } = universal approximator

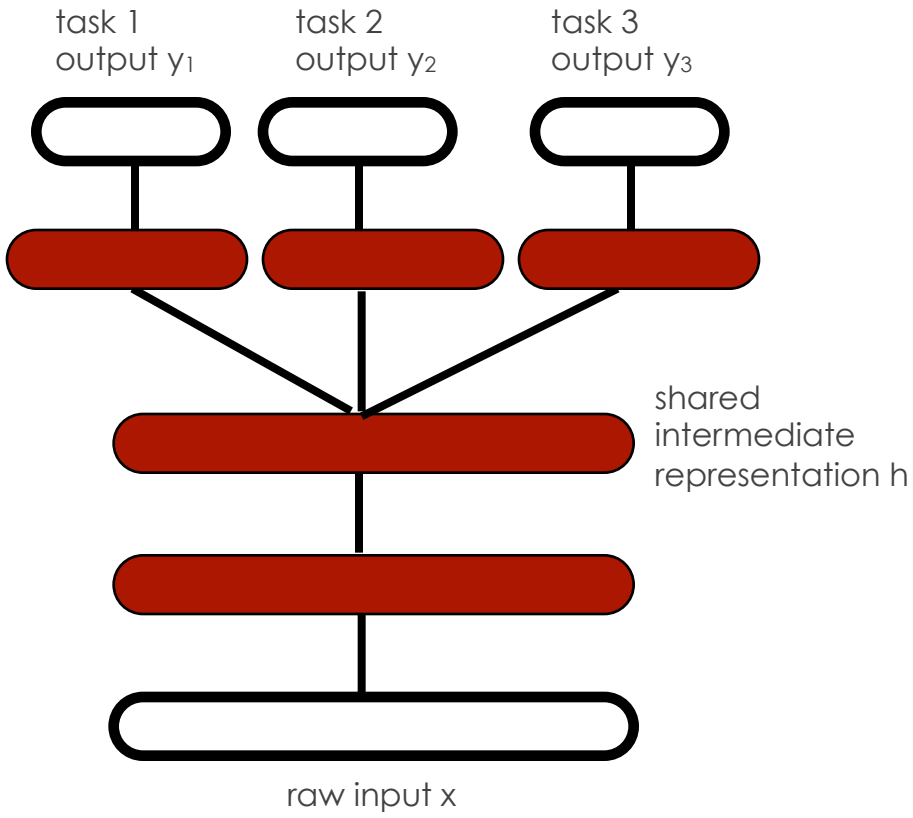


Theorems for all 3:
(Hastad et al 86 & 91, Bengio et al 2007)

Functions compactly represented with k layers may require exponential size with k-1 layers

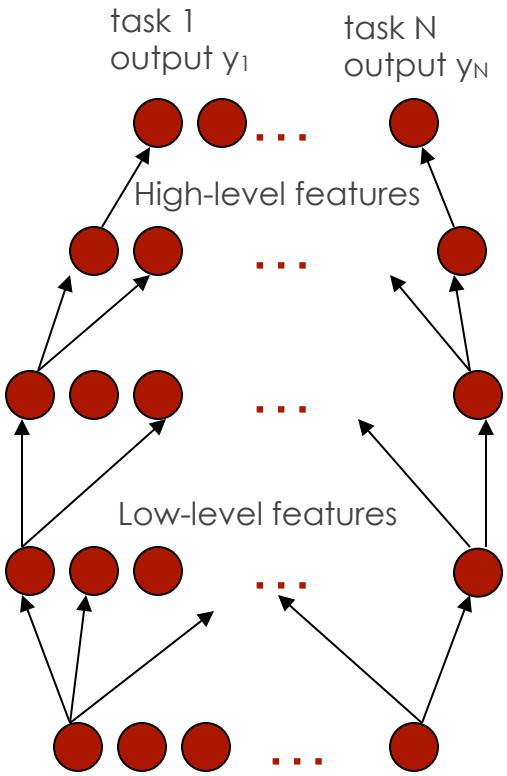
Deep Architectures and Sharing Statistical Strength, Multi-Task Learning

- Generalizing better to new tasks is crucial to approach AI
- Deep architectures learn good intermediate representations that can be shared across tasks
- A good representation is one that makes sense for many tasks

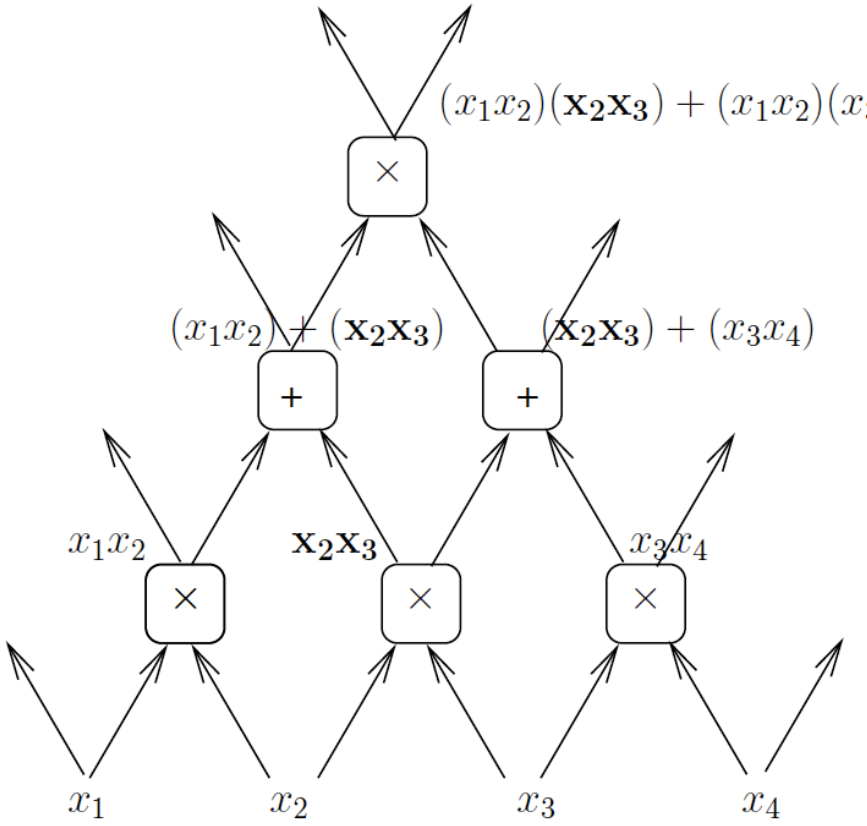


Feature and Sub-Feature Sharing

- Different tasks can share the same high-level feature
- Different high-level features can be built from the same set of lower-level features
- More levels = up to exponential gain in representational efficiency



Sharing Components in a Deep Architecture



Polynomial expressed with shared components:

advantage of depth may grow exponentially

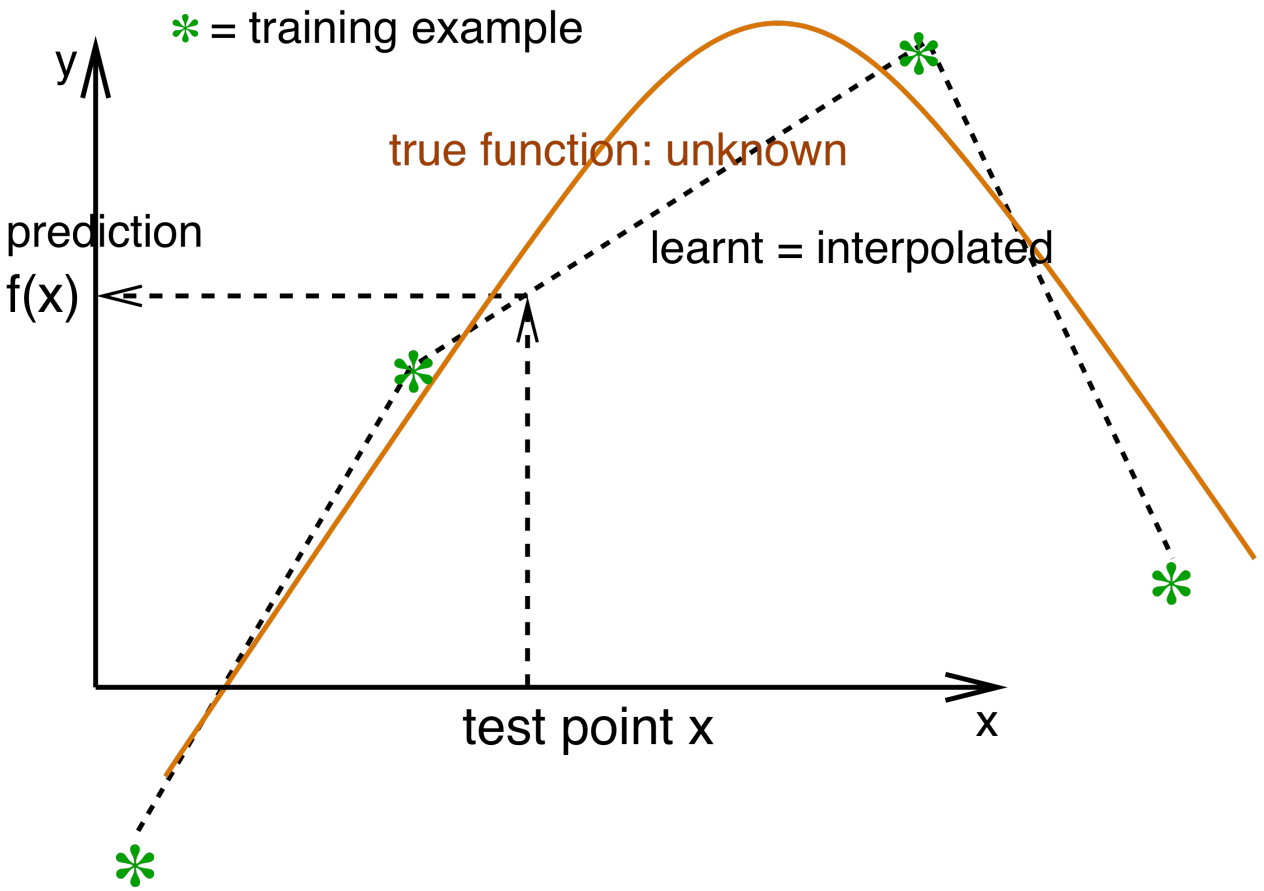
The Deep Breakthrough

- Before 2006, training deep architectures was unsuccessful, except for convolutional neural nets
- Hinton, Osindero & Teh « A Fast Learning Algorithm for Deep Belief Nets », *Neural Computation*, 2006
- Bengio, Lamblin, Popovici, Larochelle « Greedy Layer-Wise Training of Deep Networks », *NIPS'2006*
- Ranzato, Poultney, Chopra, LeCun « Efficient Learning of Sparse Representations with an Energy-Based Model », *NIPS'2006*

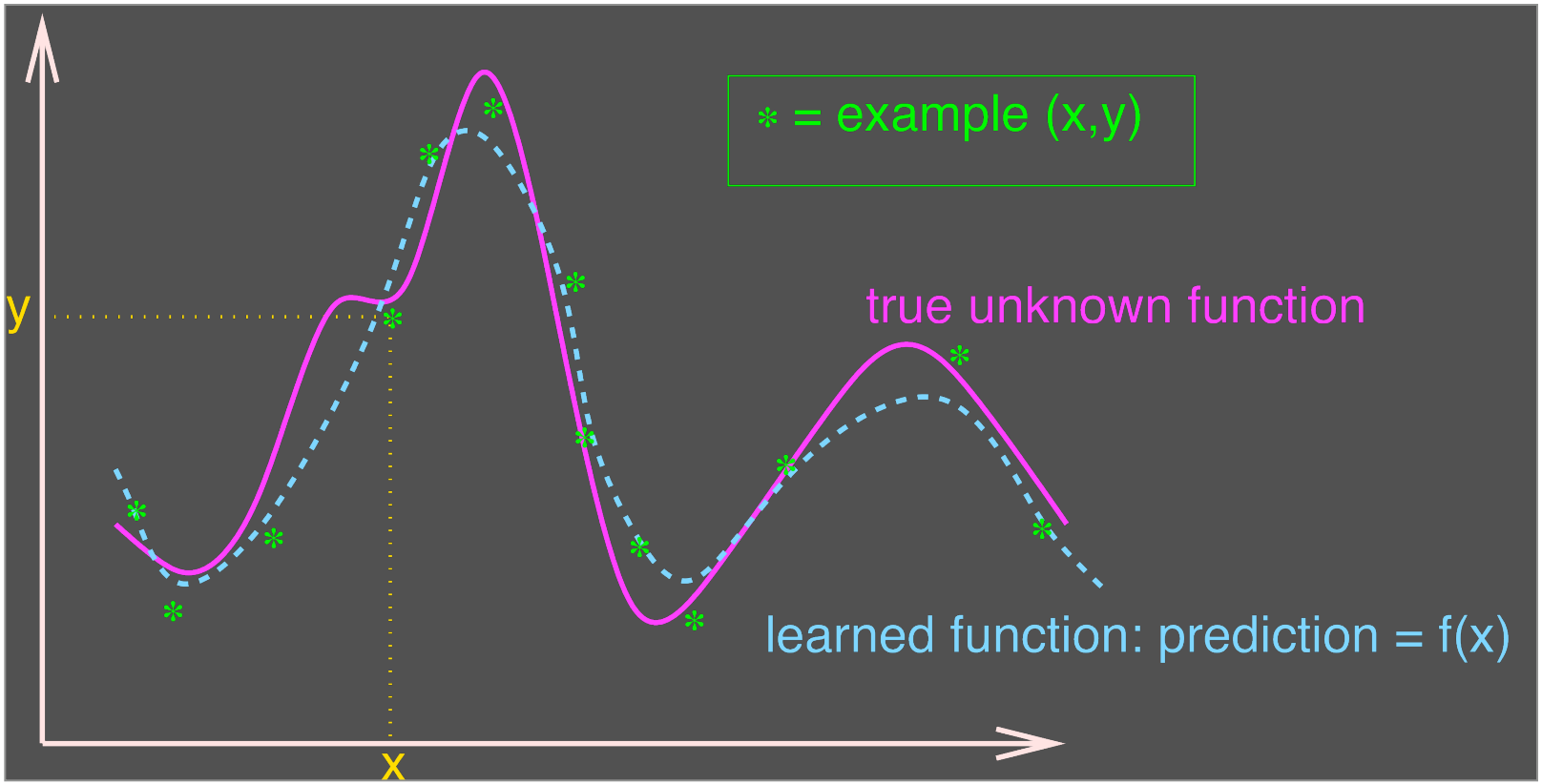
The need for non-local generalization and distributed (possibly sparse) representations

- Most machine learning algorithms are based on local generalization
- Curse of dimensionality effect with local generalizers
- How distributed representations can help

Locally Capture the Variations

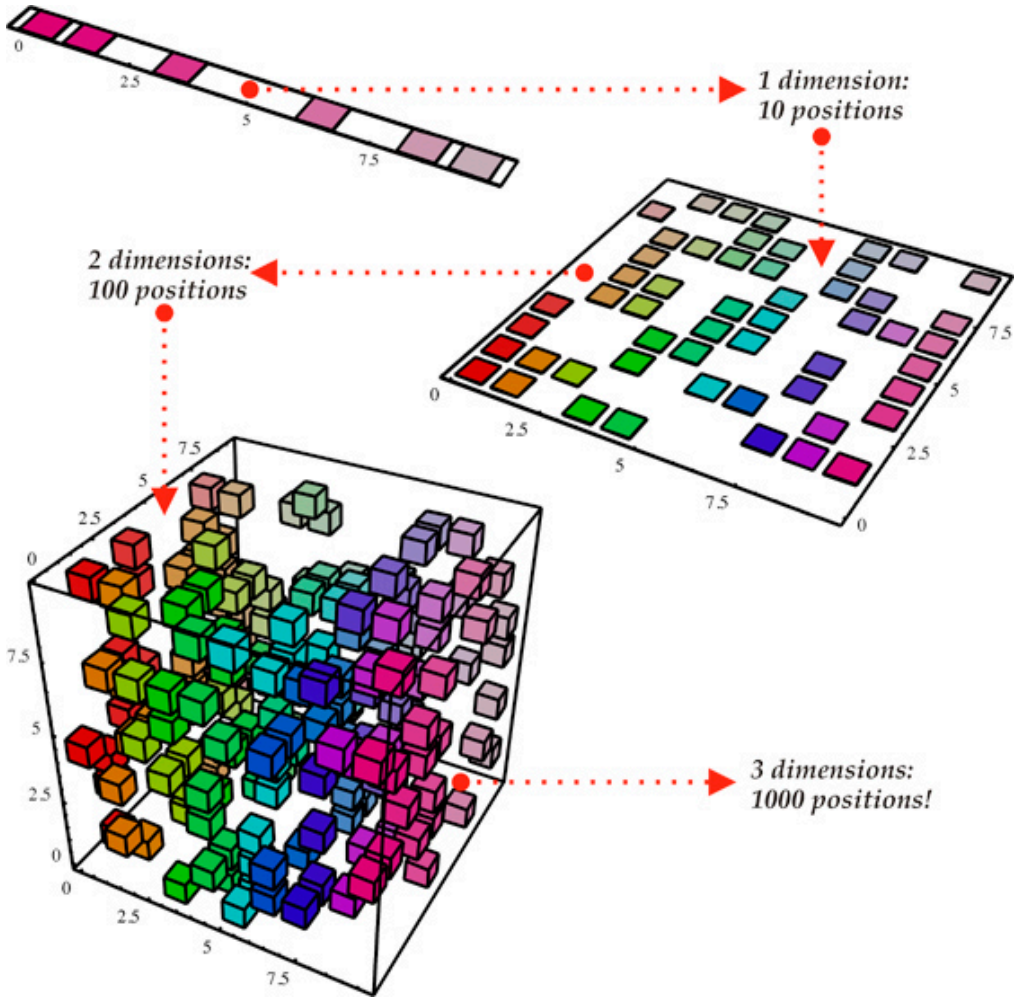


Easy with Few Variations



The Curse of Dimensionality

To generalise locally, need representative exemples for all possible variations!



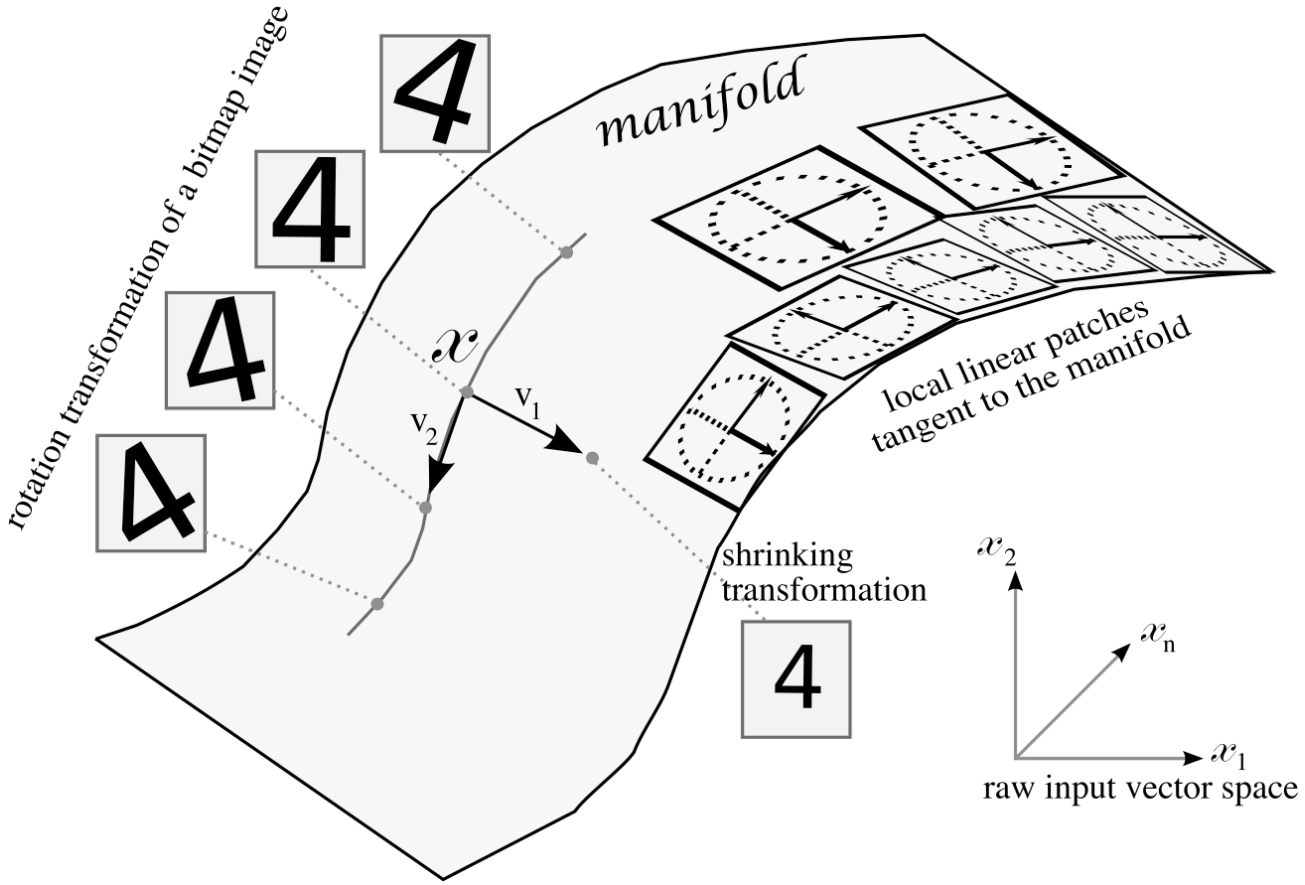
Limits of Local Generalization: Theoretical Results

(Bengio & Delalleau 2007)

- Theorem: Gaussian kernel machines need at least k examples to learn a function that has 2k zero-crossings along some line

- Theorem: For a Gaussian kernel machine to learn some maximally varying functions over d inputs require $O(2^d)$ examples

Curse of Dimensionality When Generalizing Locally on a Manifold



How to Beat the Curse of Many Factors of Variation?

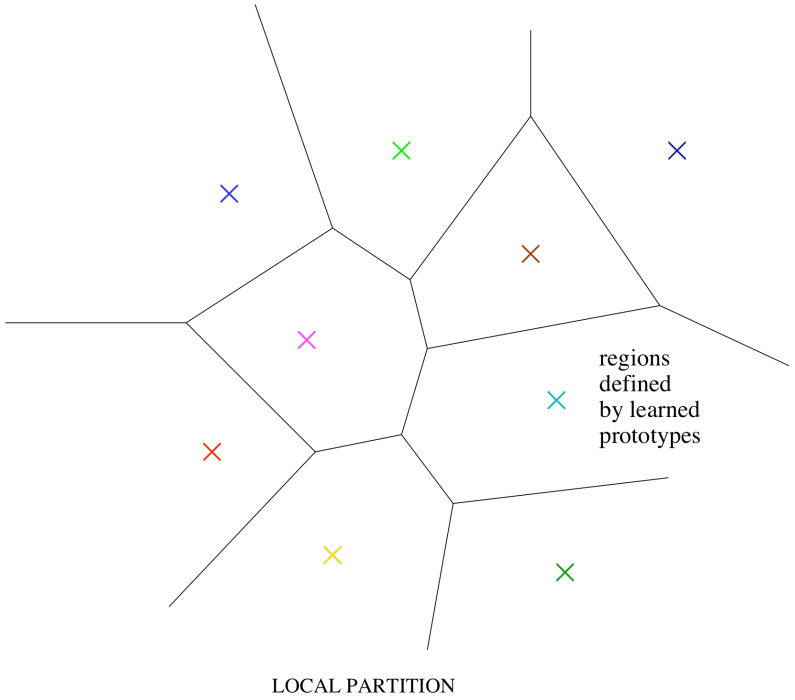
Compositionality: exponential gain in representational power

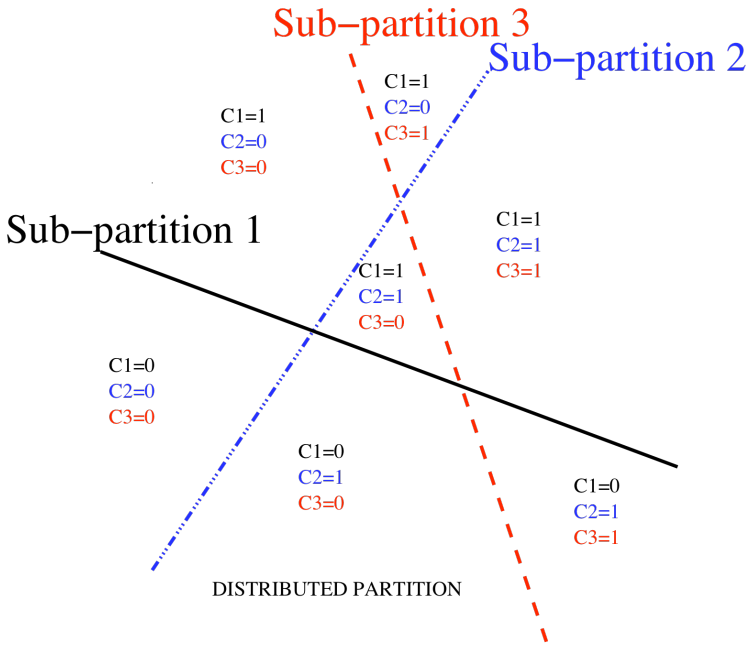
- Distributed representations
- Deep architecture

Distributed Representations
(Hinton 1986)

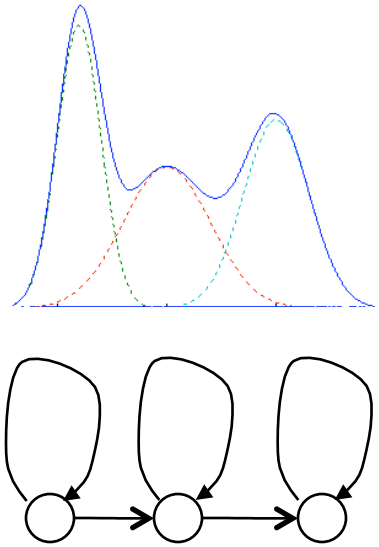
- Many neurons active simultaneously
- Input represented by the activation of a set of features that are not mutually exclusive
- Can be exponentially more efficient than local representations

Local vs Distributed

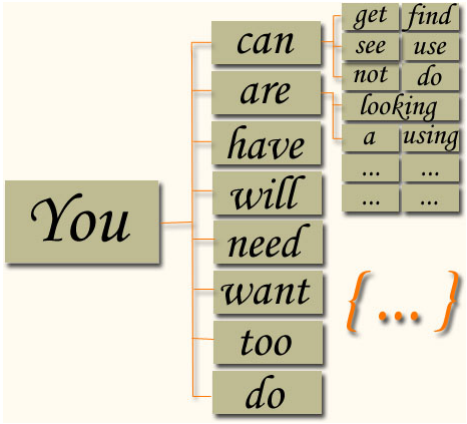




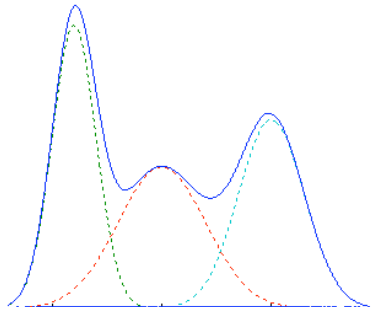
Currrent Speech Recognition & Language Modeling



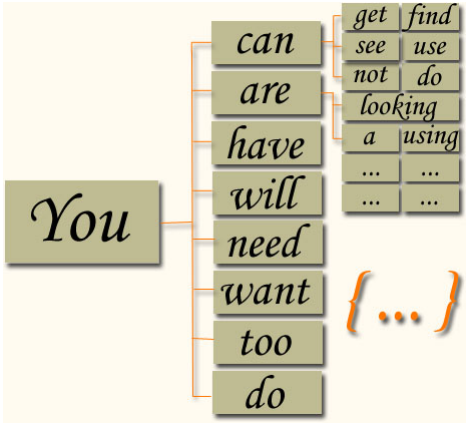
- Acoustic model: Gaussian mixture with a huge number of components, trained on very large datasets, on spectral representation
- Within-phoneme model: HMMs = dynamically warpable templates for phoneme-context dependent distributions
- Within-word models: concatenating phoneme models based on transcribed or learned phonetic transcriptions
- Word sequence models: smoothed n-grams



Current Speech Recognition & Language Modeling: **Local**

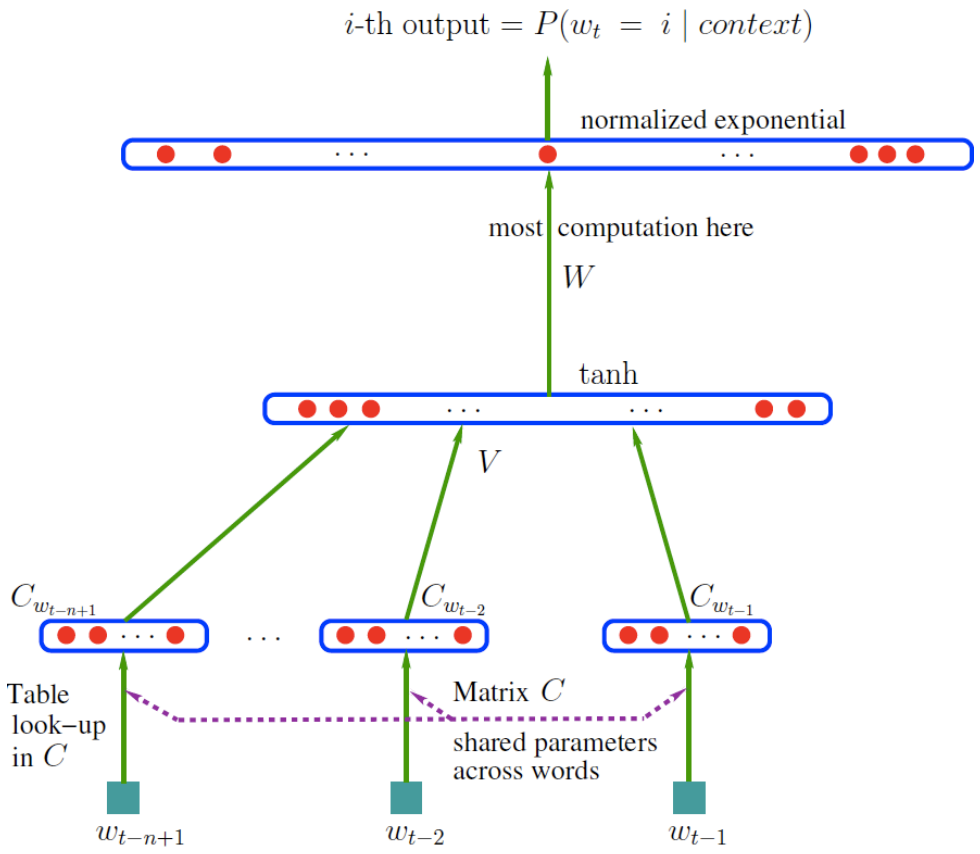


- Acoustic model: GMM = local generalization only, Euclidean distance
- Within-phoneme model: HMM = local generalization with time-warping invariant similarity
- Within-word models: exact template matching
- Word sequence models: n-grams= non-parametric template matching (histograms) with suffix prior (use longer suffixes if enough data)



Deep & Distributed NLP

- See "Neural Net Language Models" ***Scholarpedia*** entry
- NIPS'2000 and JMLR 2003 "A Neural Probabilistic Language Model"
  - Each word represented by a distributed continuous-valued code
  - Generalizes to sequences of words that are semantically similar to training sequences

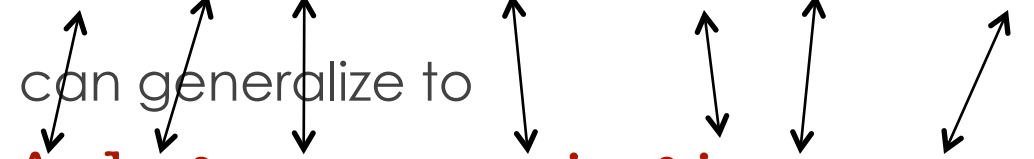


Generalization through distributed semantic representation

- Training sentence

The cat is walking in the bedroom

- can generalize to



A dog was running in a room

- because of the similarity between distributed representations for (a,the), (cat,dog), (is,was), etc.

Results with deep distributed representations for NLP

- (Bengio et al 2001, 2003): beating n-grams on small datasets (Brown & APNews), but much slower
- (Schwenk et al 2002,2004,2006): beating state-of-the-art large-vocabulary speech recognizer using deep & distributed NLP model, with ***real-time*** speech recognition
- (Morin & Bengio 2005, Blitzer et al 2005, Mnih & Hinton 2007,2009): better & faster models through hierarchical representations
- (Collobert & Weston 2008): reaching or beating state-of-the-art in multiple NLP tasks (**SRL**, POS, NER, chunking) thanks to unsupervised pre-training and multi-task learning
- (Bai et al 2009): ranking & semantic indexing (info retrieval).

Thank you for your attention!

- Questions?
- Comments?