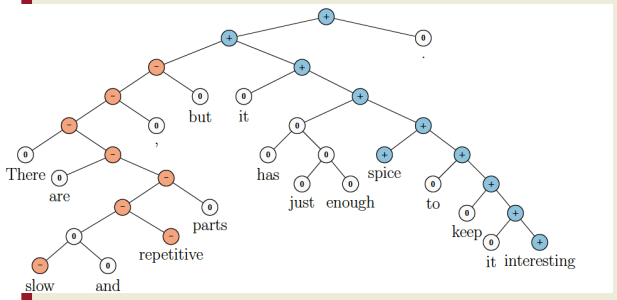
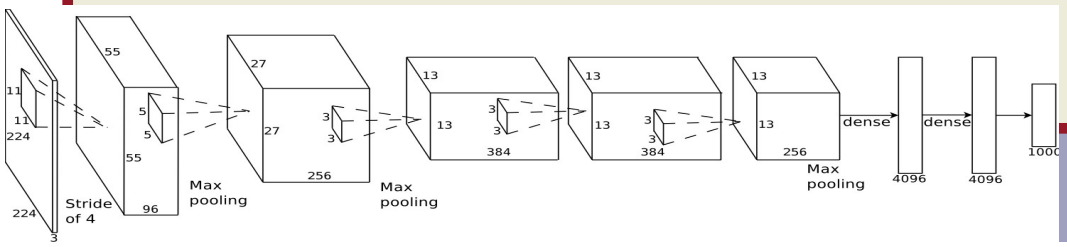
Deep Learning

for sentiment analysis





Based on slides from C.J. Taylor, Alex Krizhevsky, Ilya Sutskever, Geoff Hinton, Jeff Dean, Richard Socher, Lyle Ungar ...

Deep learning is taking over

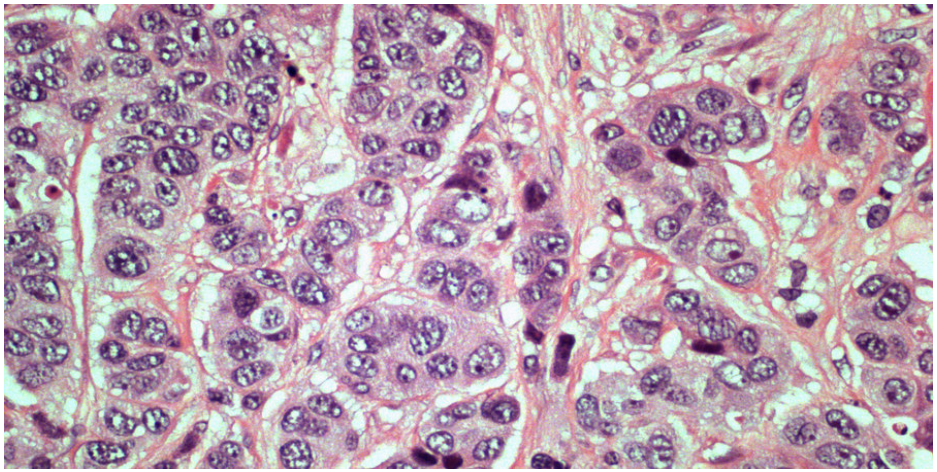
- **Face/Object/Scene recognition**
  - Self driving cars
- **Speech recognition ("speech to text")**
  - Siri, …
- **Machine translation**
  - Google translate
- **Sentiment analysis?**

Deep learning core idea

- **y = f(x)**
  - Where we don't know the functional form of f(x)
- **Given vast amounts of labeled training data we should be able use a very flexible model to fit f(x;w)**
- **Sentiment analysis sometimes fits this form**
  - y = product rating
  - **x** = review

Flexible model forms

X	y
biopsy image	Cancer present?



Flexible model forms

X
Camera image

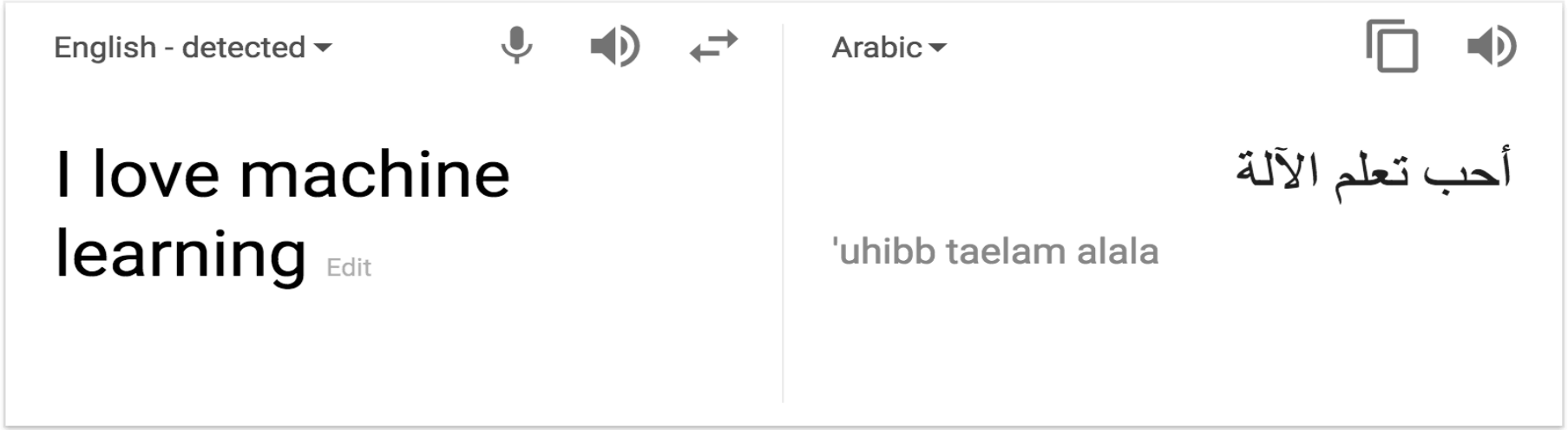
y
Objects in it

nvidia

Flexible model forms

X
English sentence

y
Translation



Artificial Neural Nets

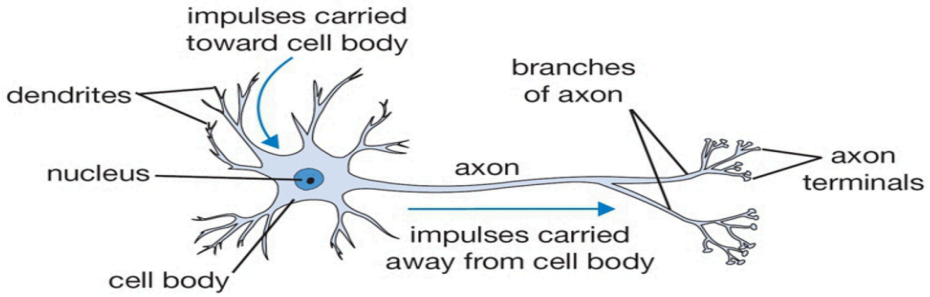
- **Non-parametric**
  - Or, technically, semi-parametric
  - Flexible model form
- **Used when there are vast amounts of data**
  - Hence popular (again) now
  - But recently with slightly smaller training sets.
- **Deep networks**
  - Idea: representation should have *many* different levels of abstraction

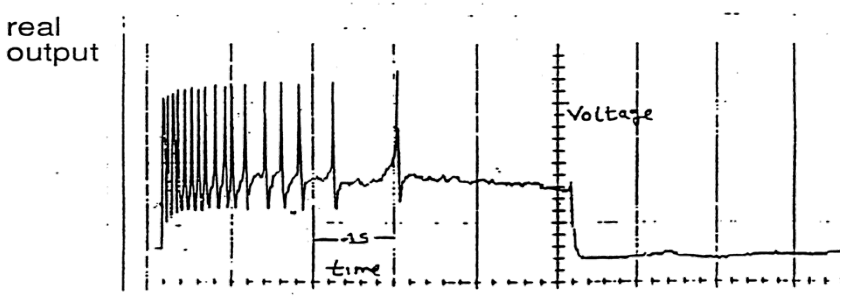
Neural Nets can be

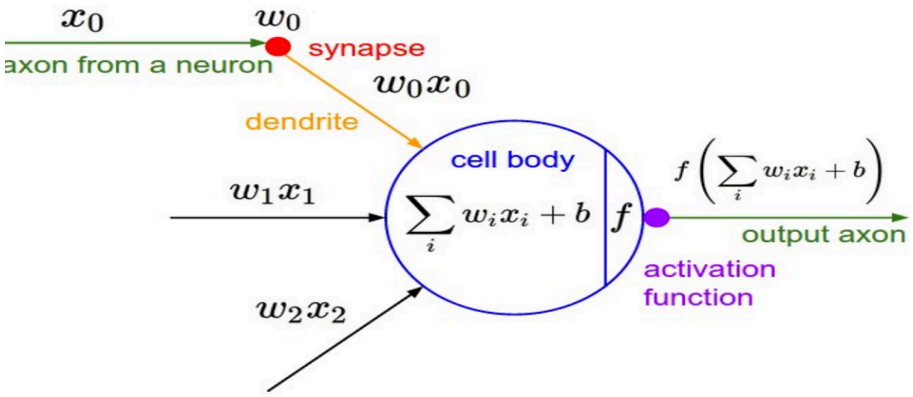
- **Supervised**
  - Generalizes *logistic regression* to a semi-parametric form
- **Unsupervised**
  - Generalizes *PCA* to a semi-parametric form

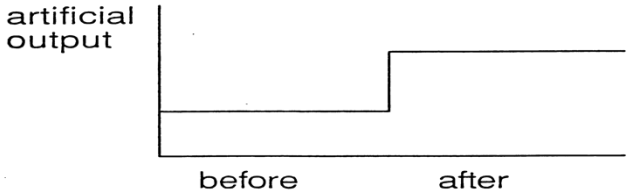
Neural nets often have built in structure

"Real" and Artificial neuron









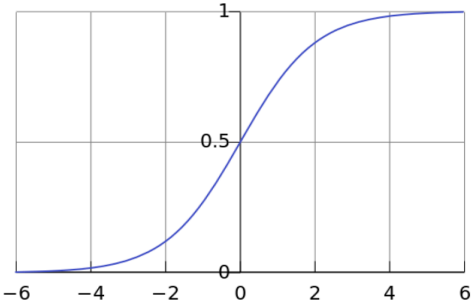
http://cs231n.github.io/neural-networks-1/

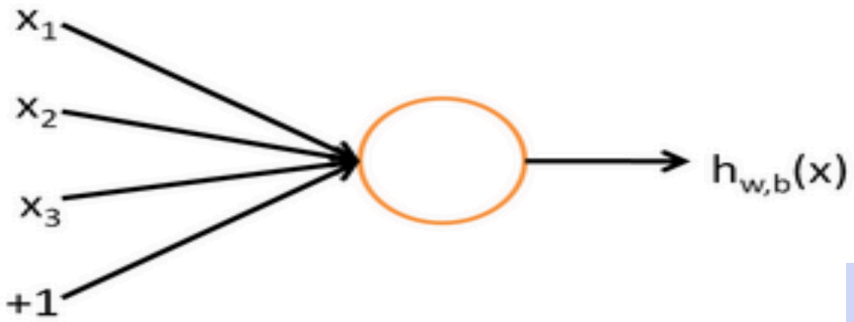
One neuron does logistic regression

$$h_{w,b}(x) = f(w^{\mathsf{T}}x + b)$$

b: We can have an "always on" feature, which gives a class prior, or separate it out, as a bias term

$$f(z) = \frac{1}{1+e^{-z}}$$

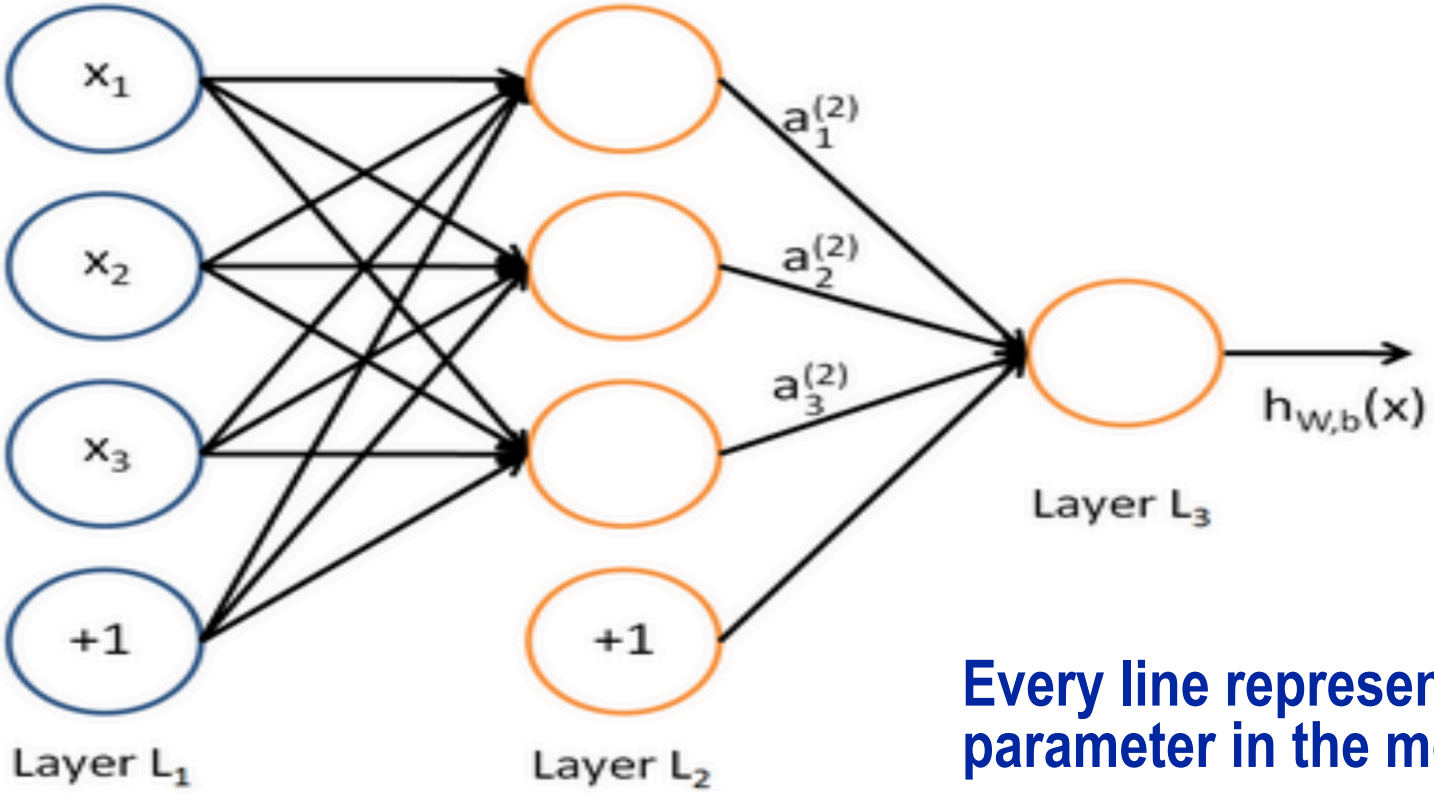




w, b are the parameters of this neuron i.e., this logistic regression model

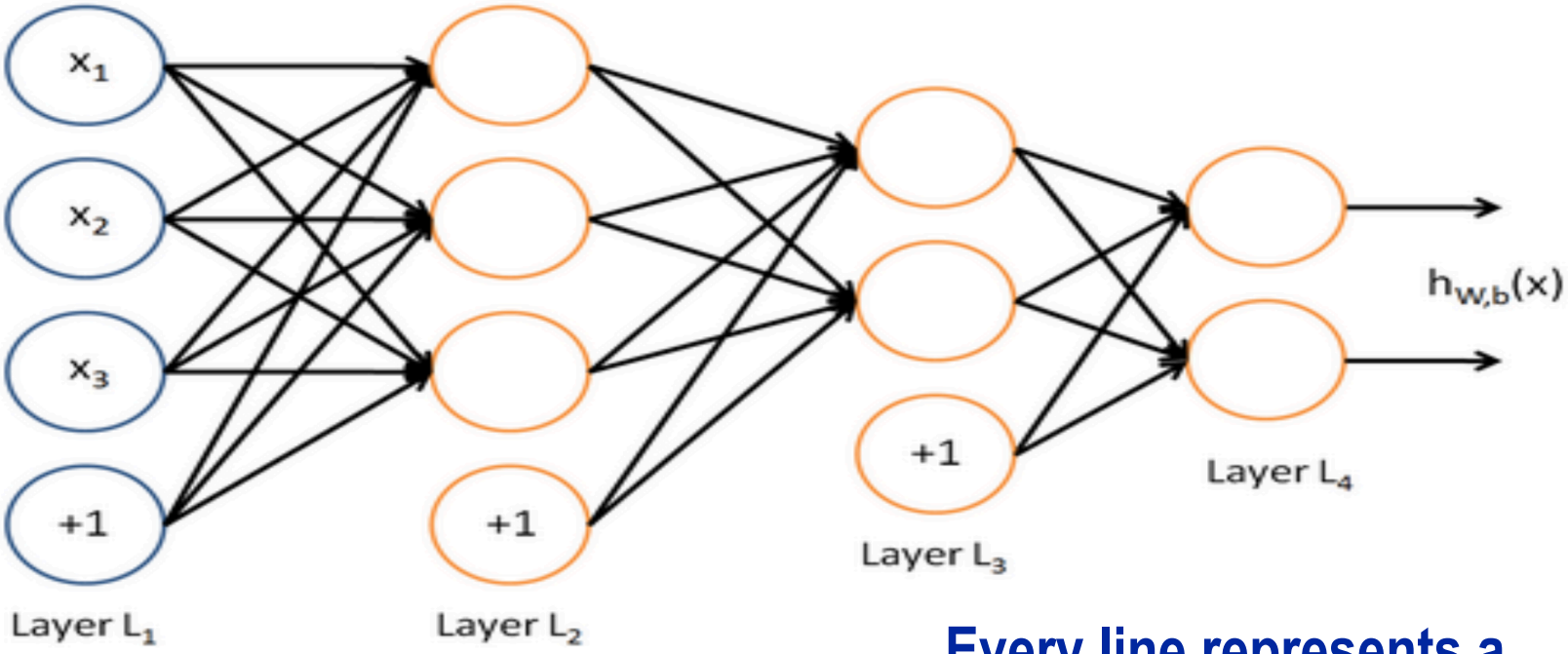
Socher and Manning tutorial

Neural Nets stack logistic regressions



Every line represents a parameter in the model

Neural Nets stack logistic regressions



Every line represents a parameter in the model, estimated using gradient descent

ANNs do pattern recognition

- **Map input "percepts" to output categories or actions**
  - Image of an object → what it is
  - Image of a person → who it is (or how they are feeling)
  - Picture → caption describing it (or the sentiment it evokes)
  - A word → the sound of saying it (or the sentiment it evokes)
  - Sound of a word → the word
  - Sequence of words in English → their Chinese translation
- **Special ANNs for handling time series**
  - E.g. sequences of words

Street View House Numbers

- Classify 32x32 color images of digits
- Digits taken from housenumbers in Google Street View
- **Train:** 604,388
- **Test:** 26,032

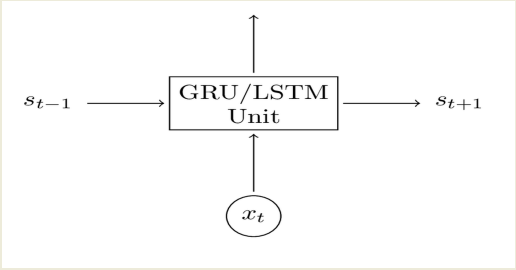
Error (%)	Method	Reference
36.7	WDCH	Netzer et al. (2011)
15	HOG	Netzer et al. (2011)
9.4	KNN	Netzer et al. (2011)
2.47	conv-DNN	Goodfellow et al. (2013)
2	Human	Netzer et al. (2013)
1.92	conv-DNN	Lee et al. (2015)

Static Deep Learning

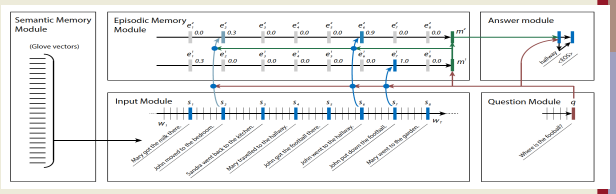
- **Supervised neural nets generalize *logistic regressions of logistic regressions***
  - First layers do data-driven feature transformations
  - Later layers find "deeper" features
- **Estimated using "minibatch" stochastic gradient descent plus chain rule ("backpropagation")**
- **Often used together with unsupervised neural nets**
  - Pretraining or dimensionality reduction

For image recognition, neural nets often have built in structure – local receptive fields and max-pooling

Recurrent Neural Networks



Time Series



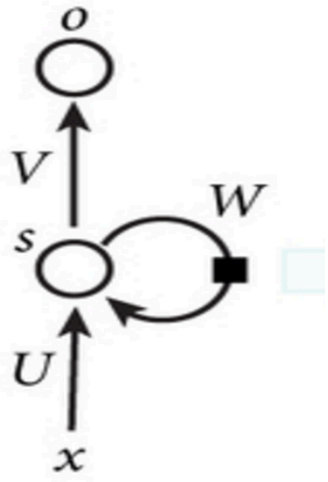
Lyle Ungar, University of Pennsylvania

Recurrent Neural Nets

- **Generalize HMMs or Linear Dynamical Systems**
  - Hidden state dynamical models, but *nonlinear*
- **Needed if you have inputs of varying length**
  - E.g. sequence of observations
    - speech
    - text
    - robots
    - videos

Simple Recurrent Neural Net

$$s_t = \tanh(Ux_t + Ws_{t-1})$$
$$o_t = \mathrm{softmax}(Vs_t)$$



x_t = input (e.g. a word)
s_t = hidden state
o_t = output (e.g. probability of the next word)
y_t = true value (e.g. **x_{t+1}**)

Softmax $\sigma(\mathbf{z})$ transforms the K-dimensional real valued output **z** to a distribution – *like logistic regression*

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for } j = 1, \ldots, K.$$

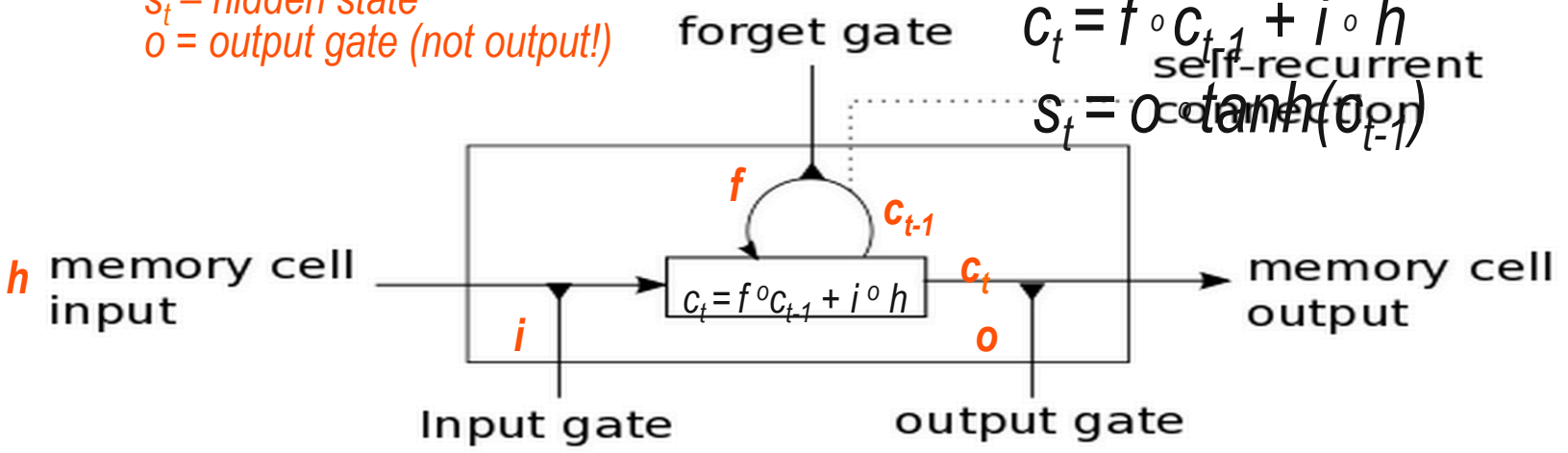
History is forgotten with an exponential decay

Long Short Term Memory (LSTM)

- **LSTM is a kind of gated RNN**
  - All of these gates have weights which are sigmoidal functions of weighted inputs

$$i = \sigma(U^i x_t + W^i s_{t-1})$$
$$f = \sigma(U^f x_t + W^f s_{t-1})$$
$$o = \sigma(U^o x_t + W^o s_{t-1})$$
$$h = \tanh(U^h x_t + W^h s_{t-1})$$
$$c_t = f \circ c_{t-1} + i \circ h$$
$$s_t = o \circ \tanh(c_t)$$

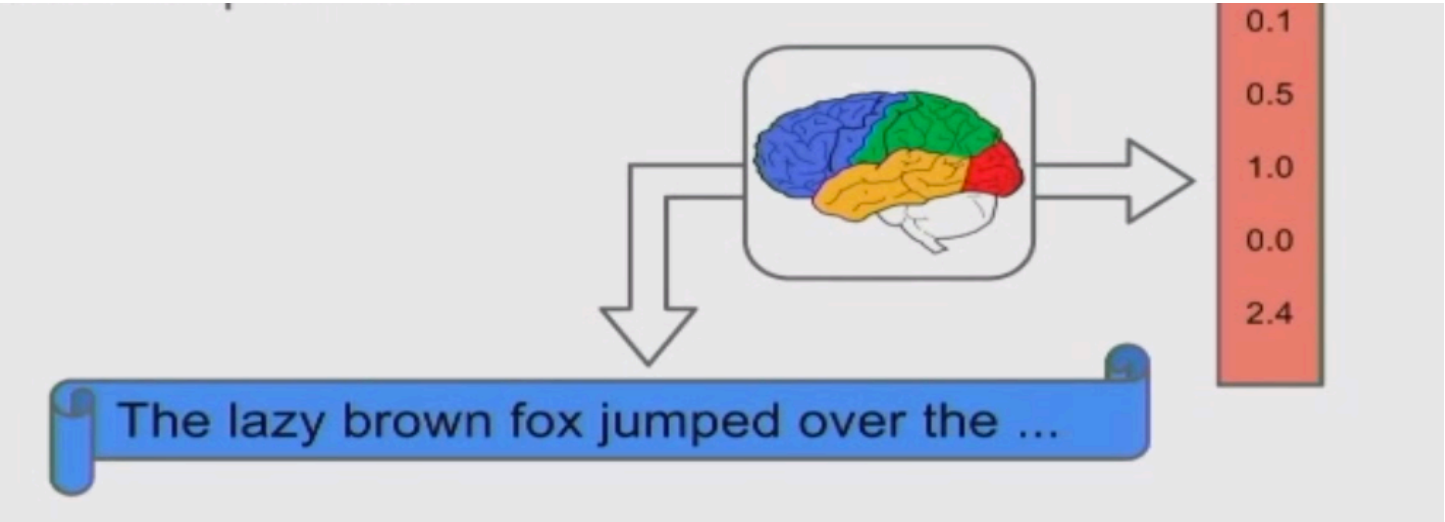
x_t – observation
s_t – hidden state
o = output gate (not output!)



Recurrent Neural Nets

- Can predict the next observation given the past observations
- Or can map one sequence to another sequence
  - An encoder
    - sentence (sequence of words) to vector
  - A decoder
    - vector to sentence (sequence of words)

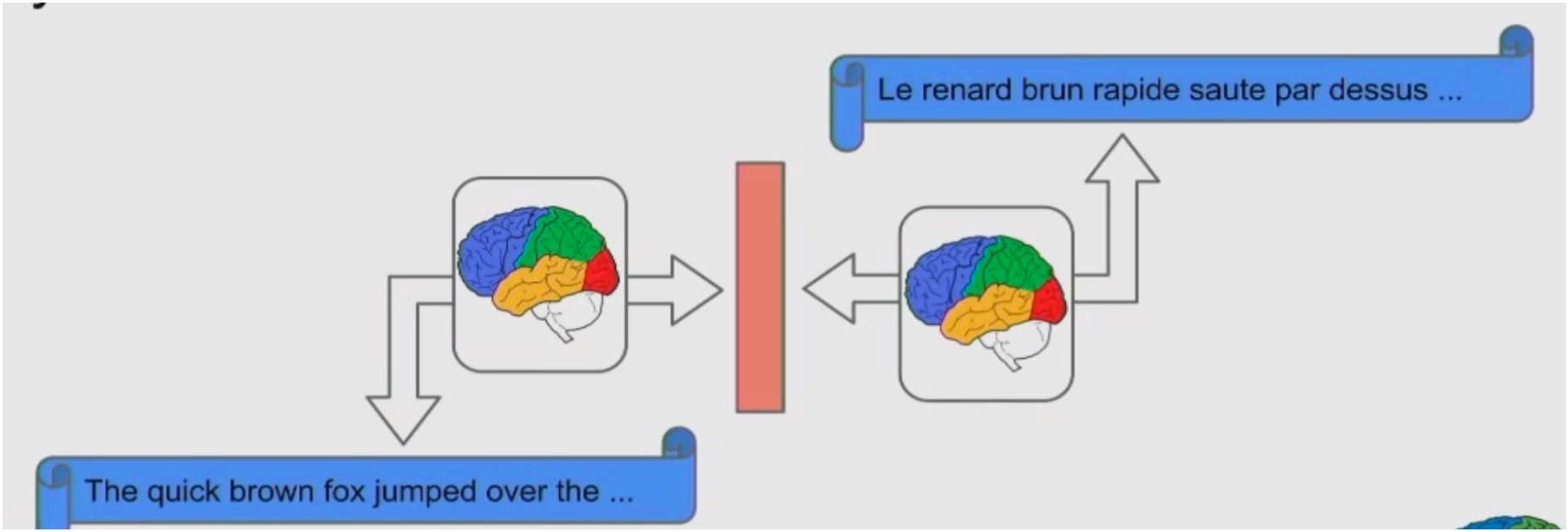
LSTM encodes a sentence



Jeff Dean, google

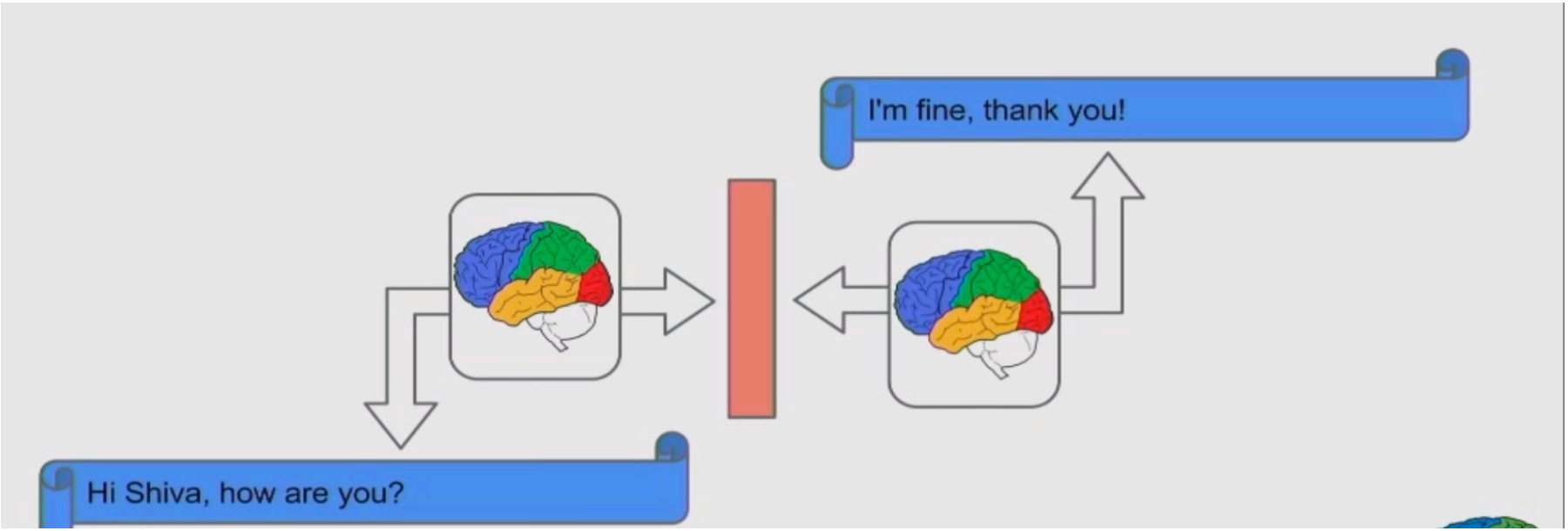
https://www.youtube.com/watch?v=90-S1M7Ny_o&spfreload=1

Encode and Decode = translate



Jeff Dean, google

… or a chatbot



Jeff Dean, google

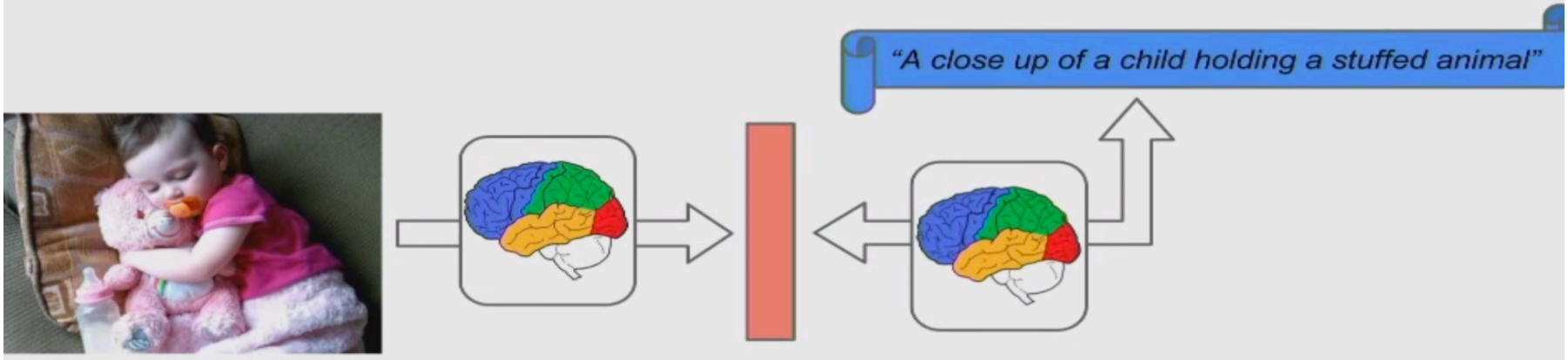
It works (sort of)

User> hello , i have a problem with my machine
BrainStop> hi
BrainStop> this is shiva .
BrainStop> how are you doing today ?
User> hi shiva , how are you ?
BrainStop> i ' m fine , thank you !
BrainStop> how may i assist you today ?
User> i want to access using vpn
BrainStop> currently the computer is connected to corp network ?
User> no
BrainStop> check this solution :
BrainStop> https : / / INTERNAL_HOSTNAME_REDACTED / articles / connecting - to - disconnecting - from - ssl - vpn # heading - 4
User> thanks , bye !
BrainStop> thank you .

Jeff Dean, google

Can map images to words

Connect sequence and image models, you get a captioning system

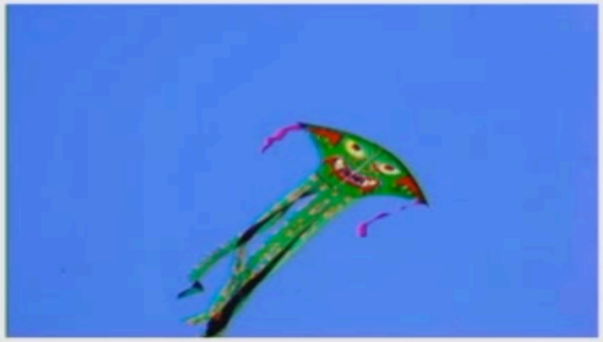


It works (sort of)

A man holding a tennis racquet on a tennis court.

Two pizzas sitting on top of a stove top oven

A group of young people playing a game of Frisbee



A man flying through the air while riding a snowboard

Input to RNN

- **Words ("one-hot")**
- **Characters ("one-hot")**
- **Word embeddings**
  - Typically 300 dimensional

Dynamic Network Summary

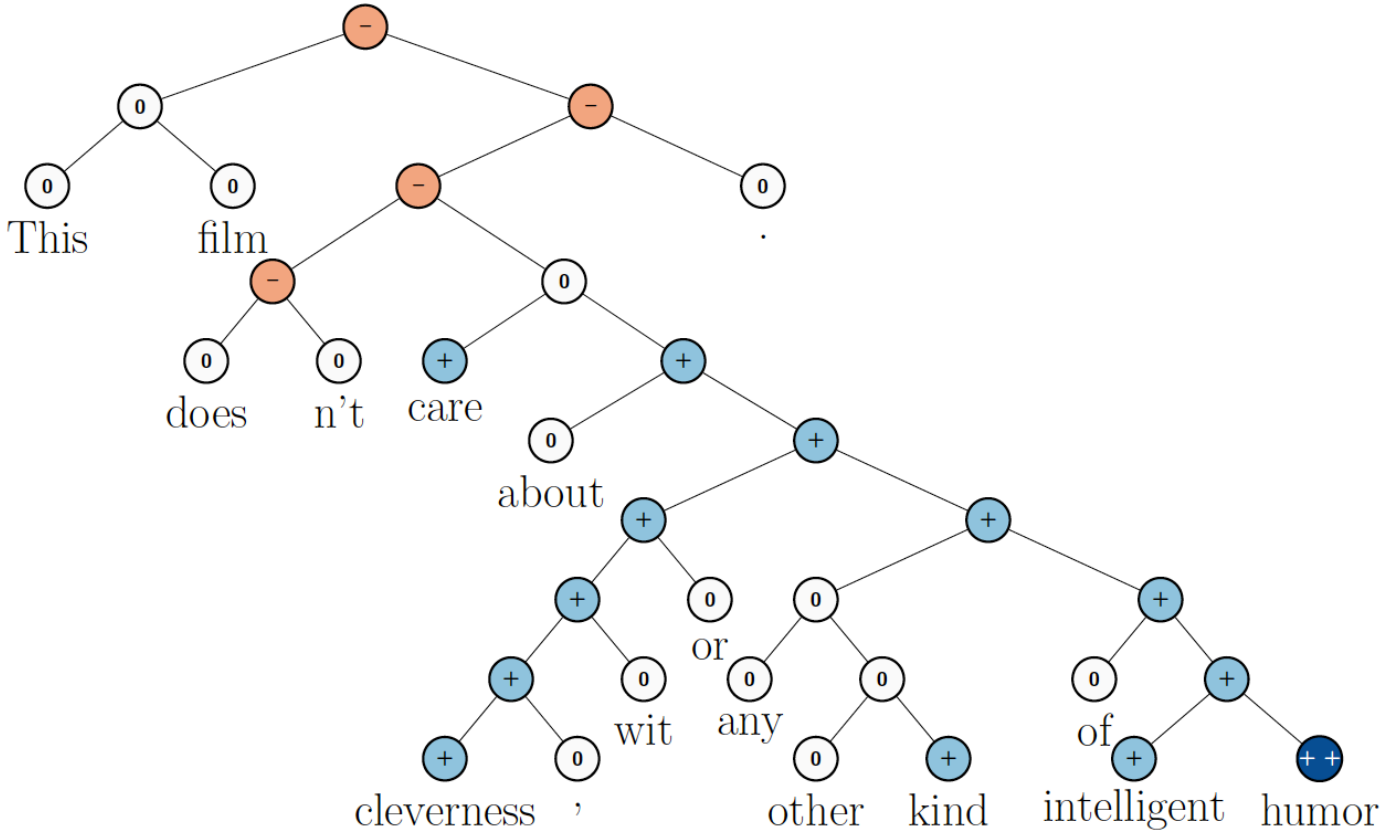
- **Gated Neural Nets/LSTMs generalize HMMs**
  - Dominate speech to text and machine translation
- **Lots of black magic "engineering"**
  - Unclear what matters about the network structure
    - depth, loss functions, regularization?
    - gating forms (LSTM, GRNN …)?
    - attention methods?
    - gradient descent
- **Good software: tensorflow, theano …**
- **They are starting to be used in sentiment analysis**
  - In a spoken dialog system, how frustrated is the speaker?

Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank

Richard Socher, Alex Perelygin, Jean Y. Wu,
Jason Chuang, Christopher D. Manning,
Andrew Y. Ng and Christopher Potts

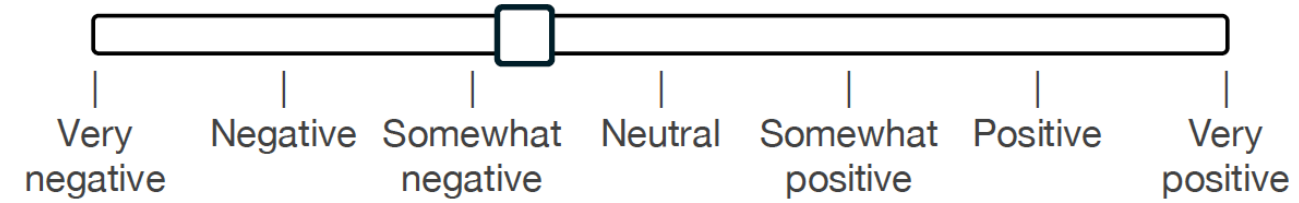
Improves accuracy from 80% up to 85.4%.

Parse tree for sentiment

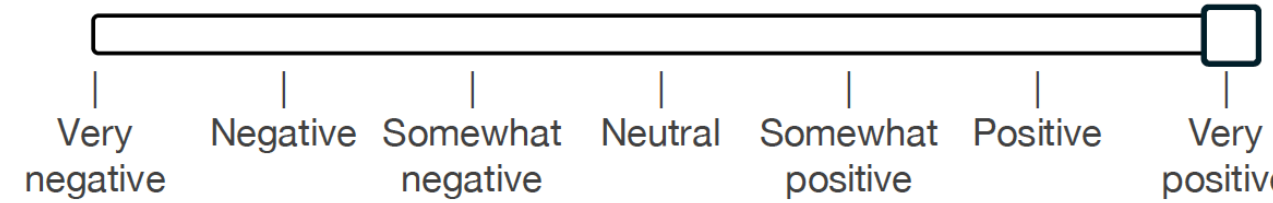


Training: label phrases

nerdy folks

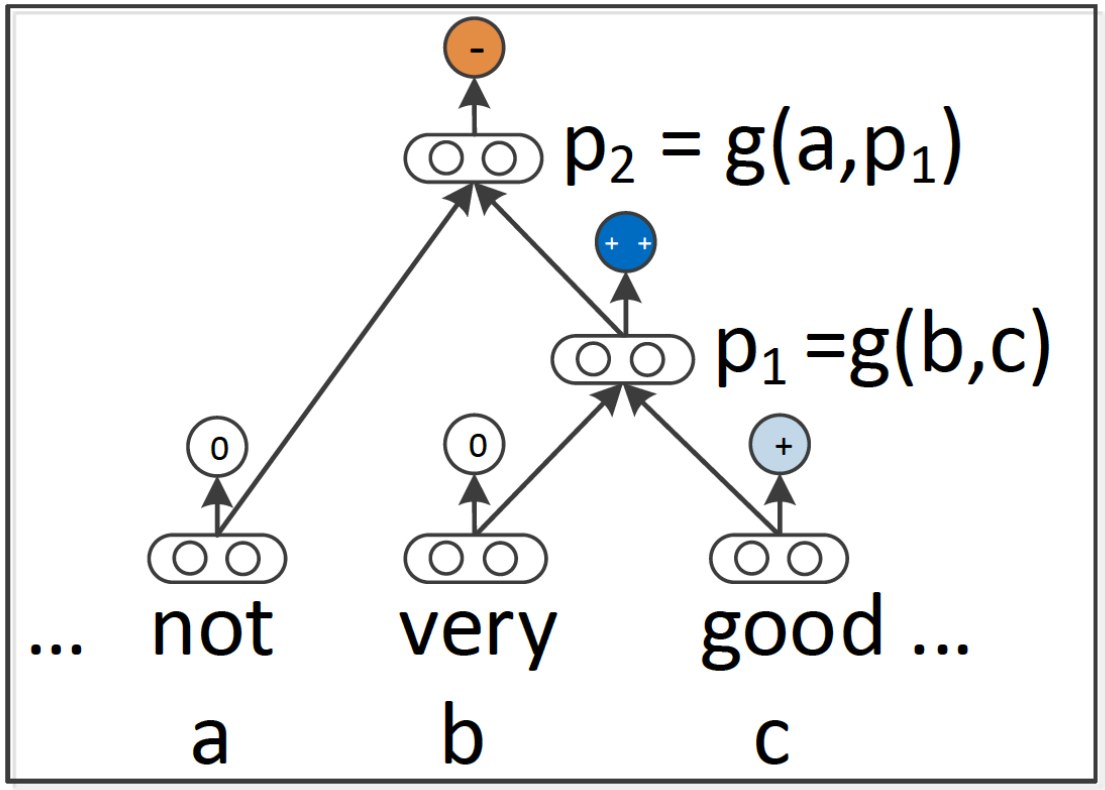


phenomenal fantasy best sellers



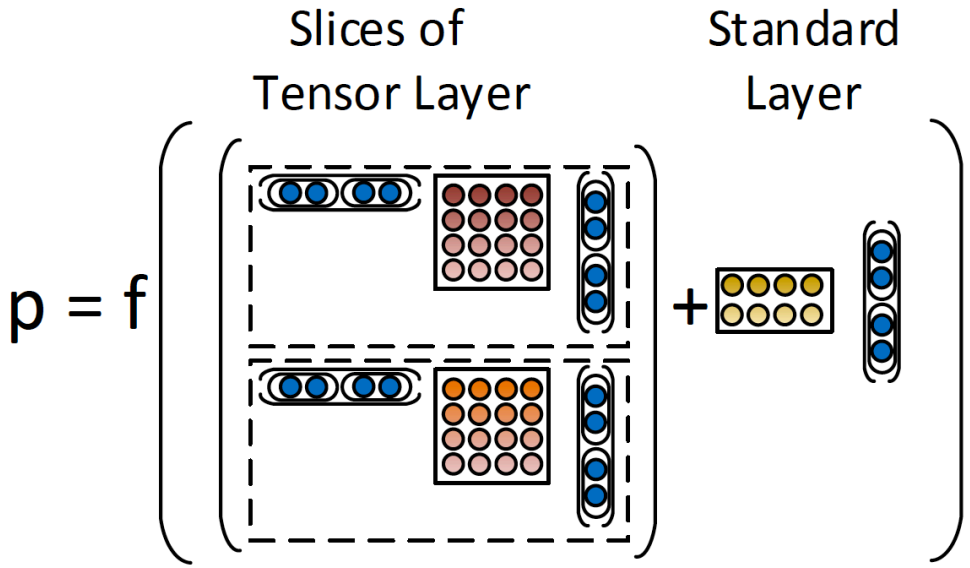
Really uses negative, somewhat negative, neutral, somewhat positive, positive

Learn recursive neural net



Compute parent vectors in a bottom up fashion using a compositionality function g and use node vectors as features for a classifier at that node.

Single Neural Tensor Layer



$$p = f\left(\begin{bmatrix} b \\ c \end{bmatrix}^T V^{[1:2]} \begin{bmatrix} b \\ c \end{bmatrix} + W \begin{bmatrix} b \\ c \end{bmatrix} \right)$$

V weight tensor
W weight matrix
b, c word vectors

Minimize error

t = target distribution (1 for correct class 0 for others)

y = network output

Θ = network parameters **V, W,** …

$$E(\theta) = \sum_i \sum_j t_j^i \log y_j^i + \lambda \|\theta\|^2$$

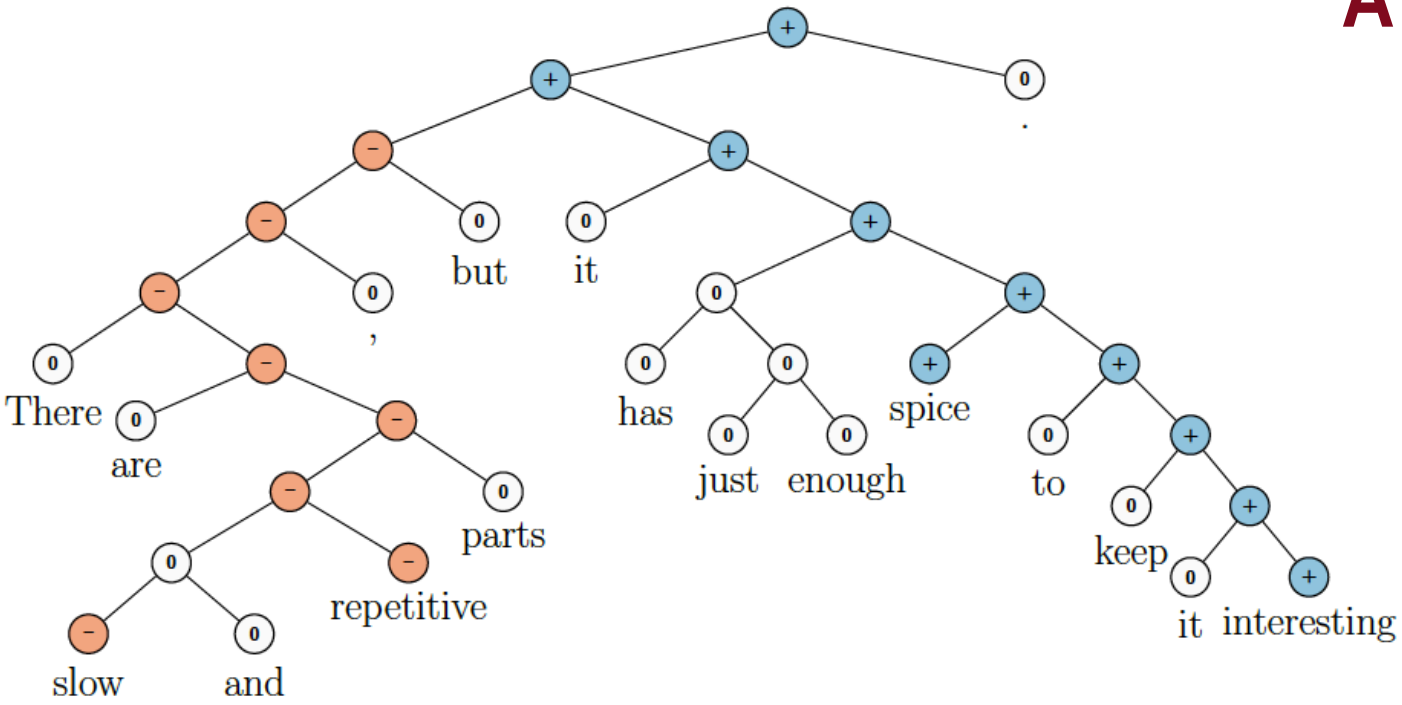
Sum over the nodes *i*, class *j*

Minimize error

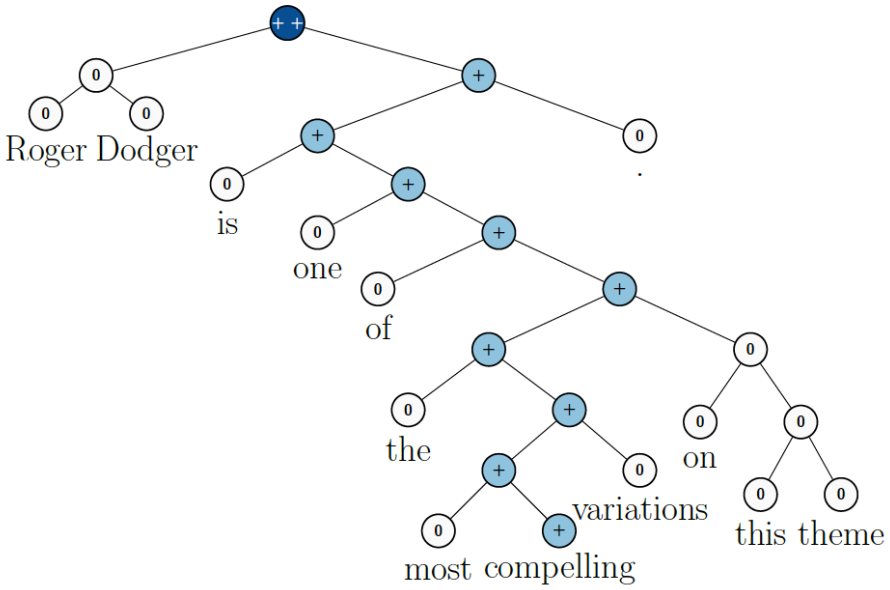
$$E(\theta) = \sum_i \sum_j t_j^i \log y_j^i + \lambda \|\theta\|^2$$

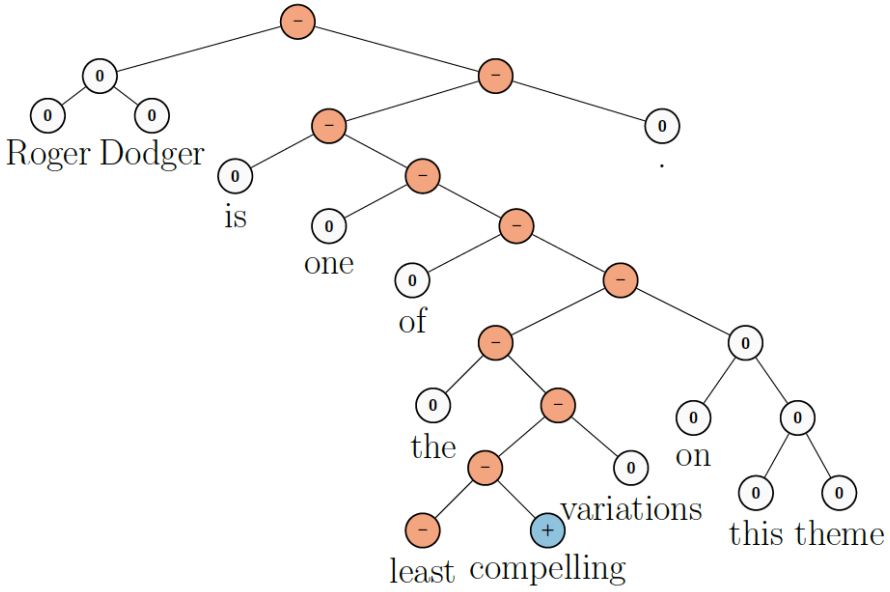
Contrastive conjunction

"A but B"

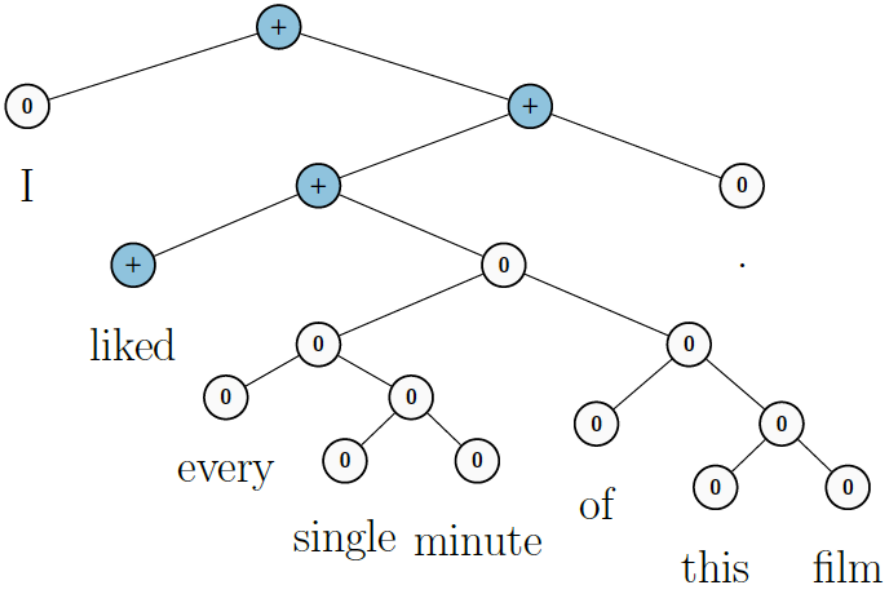


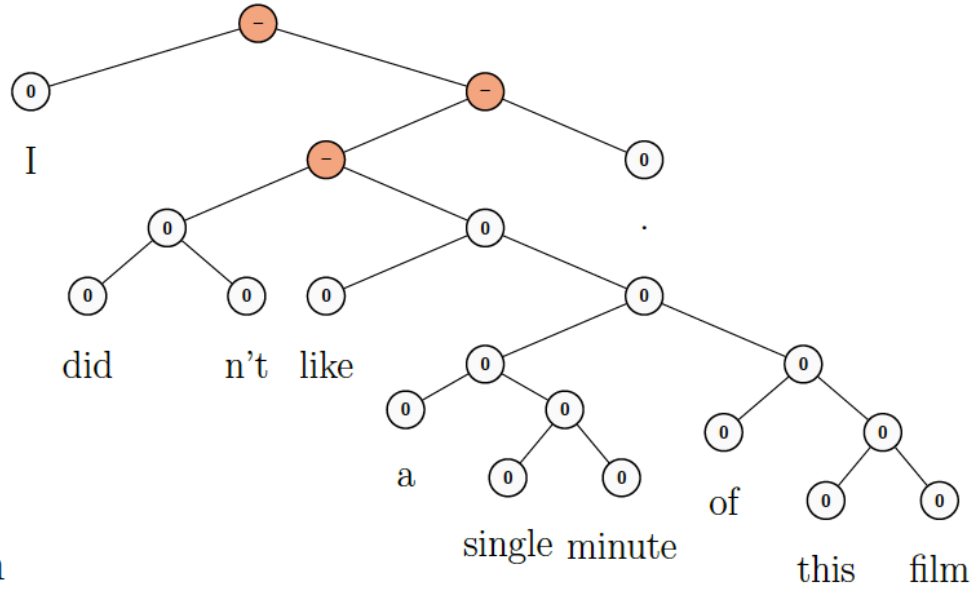
Negation


Roger Dodger
is
one
of
the
most compelling
variations
on
this theme
.


Roger Dodger
is
one
of
the
least compelling
variations
on
this theme
.

Negation


I
liked
every
single
minute
of
this
film
.


I
did
n't
like
a
single
minute
of
this
film
.

Positive and negative n-grams

Most positive n-grams	Most negative n-grams
engaging; best; powerful; love; beautiful	bad; dull; boring; fails; worst; stupid; painfully
excellent performances; A masterpiece; masterful film; wonderful movie; marvelous performances	worst movie; very bad; shapeless mess; worst thing; instantly forgettable; complete failure
an amazing performance; wonderful all-ages triumph; a wonderful movie; most visually stunning	for worst movie; A lousy movie; a complete failure; most painfully marginal; very bad sign
nicely acted and beautifully shot; gorgeous imagery, effective performances; the best of the year; a terrific American sports movie; refreshingly honest and ultimately touching	silliest and most incoherent movie; completely crass and forgettable movie; just another bad movie. A cumbersome and cliche-ridden movie; a humorless, disjointed mess
one of the best films of the year; A love for films shines through each frame; created a masterful piece of artistry right here; A masterful film from a master filmmaker,	A trashy, exploitative, thoroughly unpleasant experience ; this sloppy drama is an empty vessel.; quickly drags on becoming boring and predictable.; be the worst special-effects creation of the year

Where is deep learning going?

- **Attention models to find the relevant parts of longer documents**
  - Most sentiment analysis is done using product reviews, which implicitly label a whole review
    - Labeling all the phrases is too expensive
- **Multi-modal**
  - All image analysis uses deep learning
- **Semi-supervised learning**
  - Learn a model to e.g. recognize smiling faces
  - Then use its outputs as features for SA
    - Or use the outputs of the penultimate layer