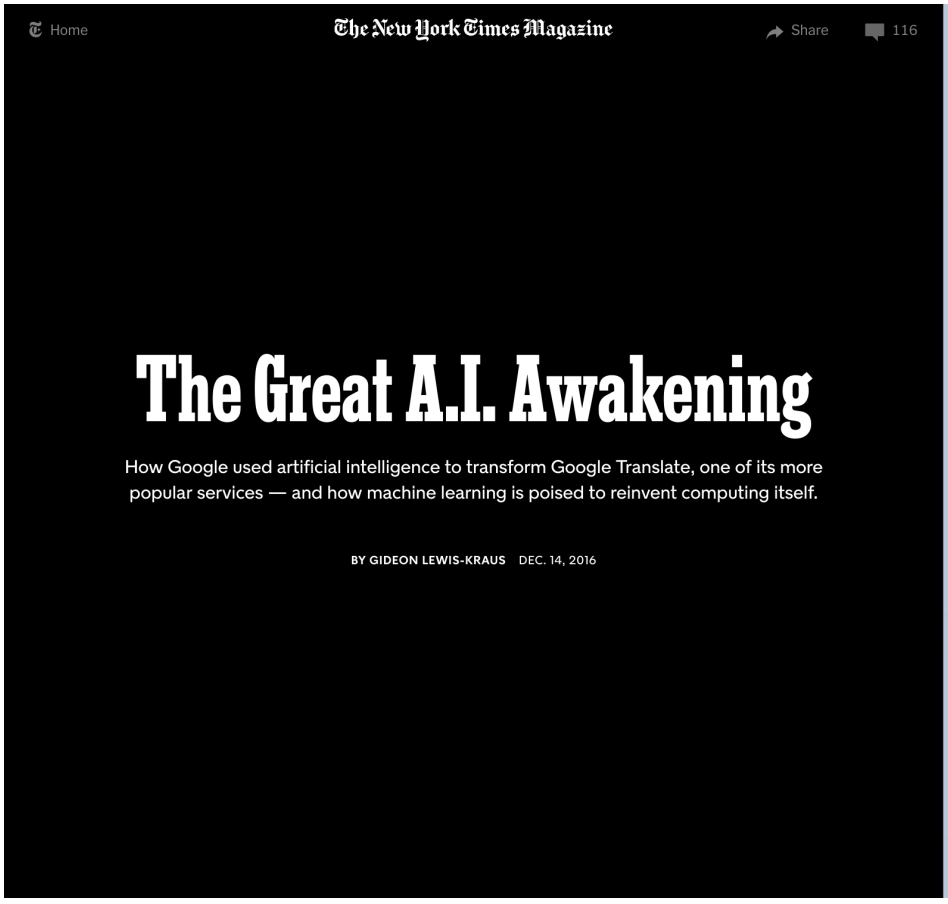
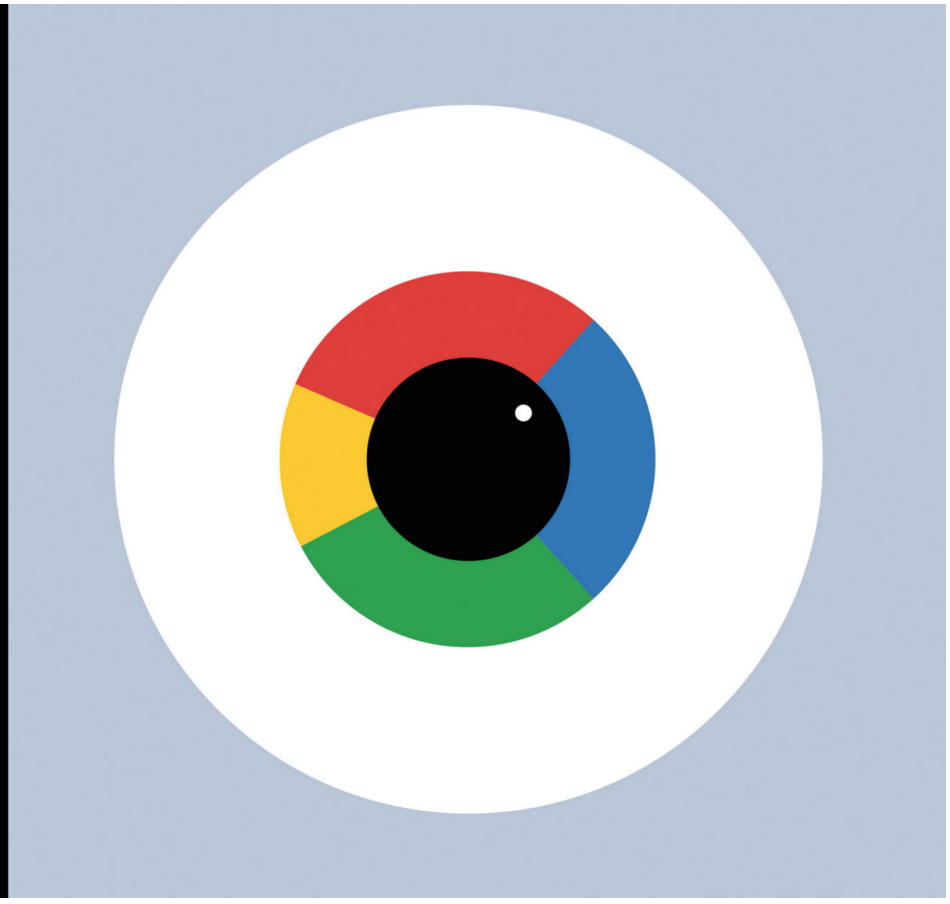
CS 109b

RNNs and Language

Alexander Rush (@harvardnlp)





(December, 2016)

Machine Translation

페루정부는유색인종과흑인의출입을달가와하지않는레스토랑에대한여러불만을들은뒤수도리마에있는유명레스토랑을닫았고, 추후통보가있을때까지임시휴업을명했다.
NMT: The government has closed a number of restaurants in the capital, Lima, after hearing complaints about a famous restaurant that does not welcome the entrance of people of color and blacks.
V8: The famous restaurant closed in Lima after hearing several complaints to the restaurant, which is not a happy access for people of color, black, Peru has ordered extra holiday until further notice.

Machine Translation

페루 국회는 인종차별적 범죄에 유죄로 수감형을 도입하는 것을 허락해서 유죄 판결을 받은 사람들에게 몇 달 전부터 이 벌금이 강력하게 적용되어 졌다

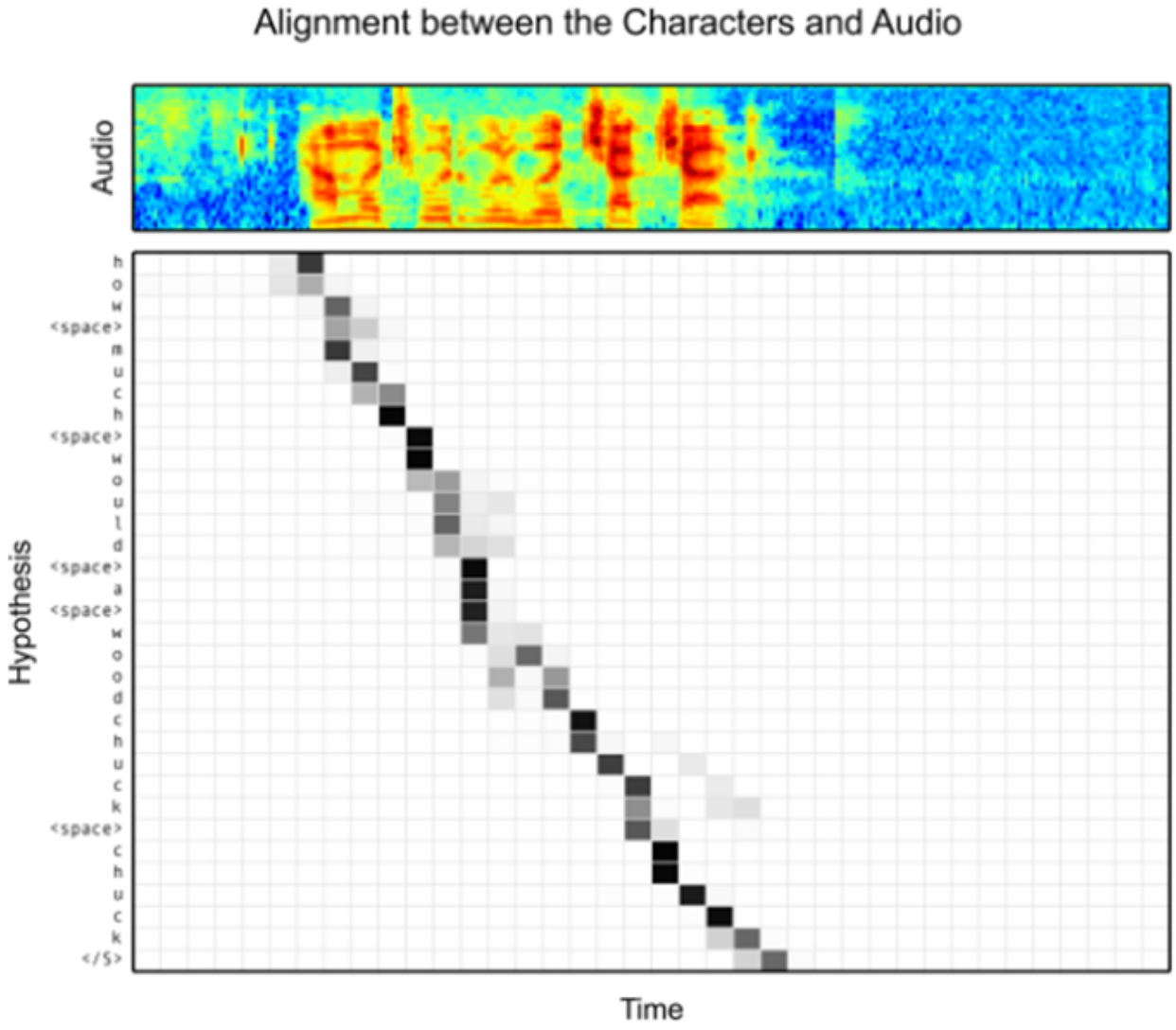
NMT: The Peruvian parliament has allowed to impose a prison sentence on those convicted of racial crimes, and a few months ago the fines were strongly applied to those convicted.

V8: Peruvian Congress allowed to introduce the arrest brother in a racist crime guilty and the penalty was applied to the convicted people from a few months ago.

Other Applications: Image Captioning (?)

(b) A woman is throwing a frisbee in a park.

Other Applications: Speech Recognition (?)



Language Models

It is a capital mistake to theorize before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts. -Sherlock Holmes, A Scandal in Bohemia

It is a capital mistake to theorize before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts. -Sherlock Holmes, A Scandal in Bohemia

It is a capital mistake to theorize before one has _____ . . .

108 938 285 28 184 29 593 219 58 772 _____ . . .

Language Modeling Task

Given a sequence of text give a probability distribution over the next word.

The Shannon game. Estimate the probability of the next letter/word given the previous.

> *THE ROOM WAS NOT VERY LIGHT A SMALL OBLONG READING LAMP ON THE DESK SHED GLOW ON POLISHED* ___

Shannon (1948) *Mathematical Model of Communication*

We may consider a discrete source, therefore, to be represented by a stochastic process. Conversely, any stochastic process which produces a discrete sequence of symbols chosen from a finite set may be considered a discrete source. This will include such cases as:

1. Natural written languages such as English, German, Chinese. ...

Shannon's Babblers

4. Third-order approximation (trigram structure as in English). IN NO 1ST LAT WHEY CRATICT FROURE BIRS GROCID PONDENOME OF DEMONSTURES OF THE REPTAGIN IS REGOACTIONA OF CRE

5. First-Order Word Approximation. Rather than continue with tetragram, ... , II-gram structure it is easier and better to jump at this point to word units. Here words are chosen independently but with their appropriate frequencies. REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT NATURAL HERE HE THE A IN CAME THE TO OF TO EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE THESE.

6. Second-Order Word Approximation. The word transition probabilities are correct but no further structure is included.

THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH 'RITER THAT THE CHARACTER OF THIS POINT IS THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE TIME OF WHO EVER TOLD THE PROBLEM FOR AN UNEXPECTED

The resemblance to ordinary English text increases quite noticeably at each of the above steps.

Goal: Estimate Markov model

$$p(w_t|w_1,\ldots,w_{t-1}) \approx p(w_t|w_{t-n},\ldots w_{t-1})$$

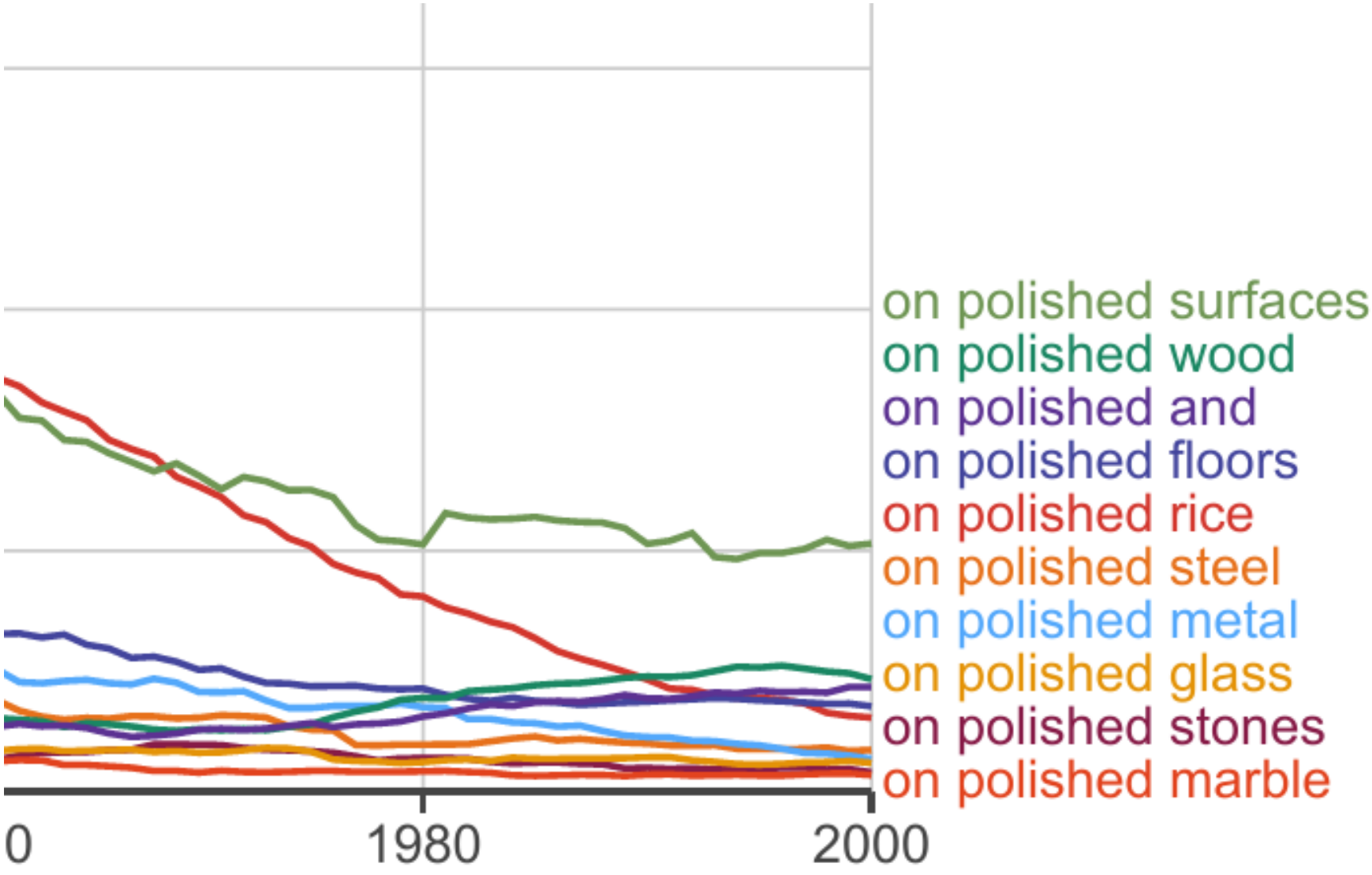
Ingredients:

- 1 Corpus (e.g. the entire web)

Steps:

- (1) Collect words, (2) Count up n-grams, (3) Divide*

$$\begin{aligned} p(w_{t+1}|w_{t-n+1},\ldots w_t) &= \frac{\#(w_{t-n+1},\ldots,w_{t+1})}{\#(w_{t-n+1},\ldots,w_t)} \\ &= \frac{\#(\text{theorize before one has data})}{\#(\text{theorize before one has})} \end{aligned}$$

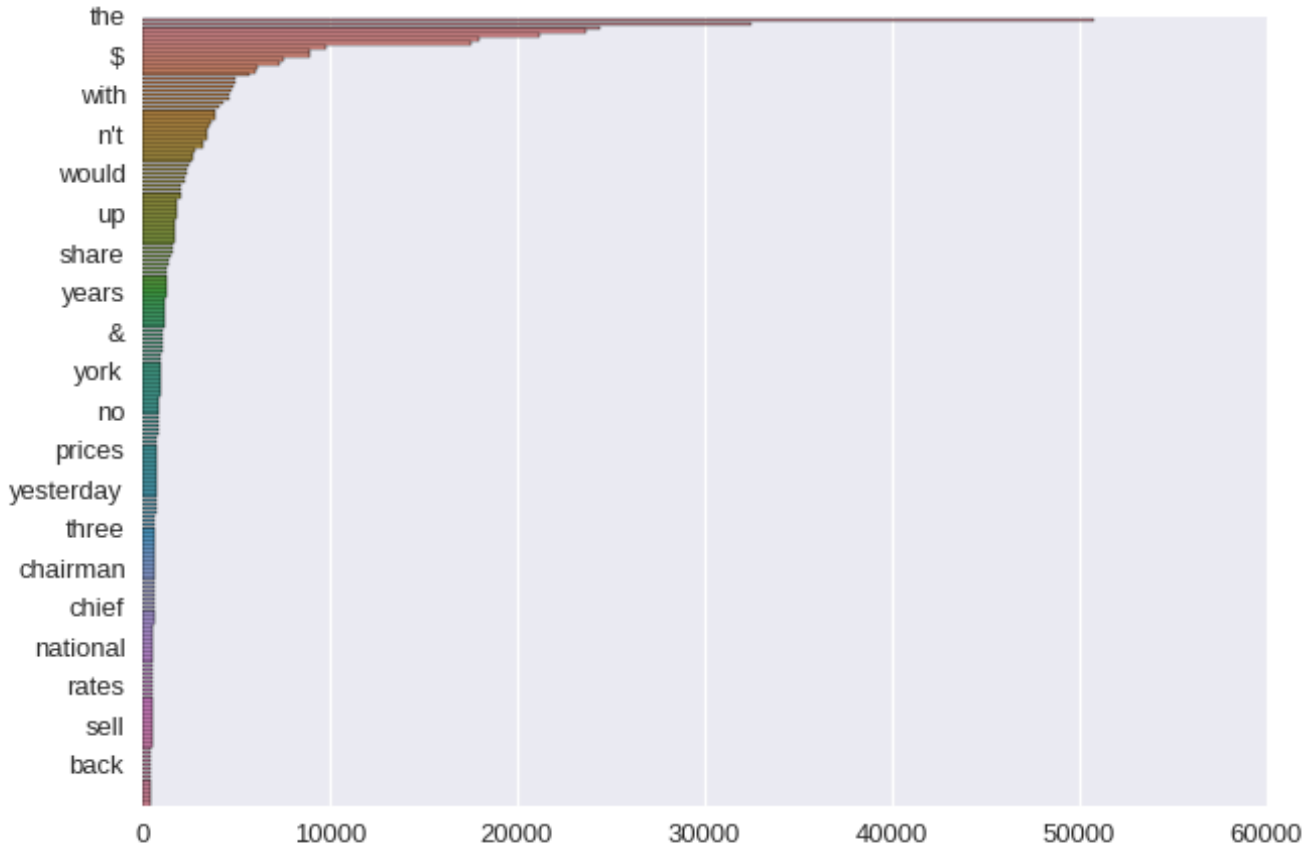

on polished surfaces
on polished wood
on polished and
on polished floors
on polished rice
on polished steel
on polished metal
on polished glass
on polished stones
on polished marble
0
1980
2000

Google 1T

Number of token	1,024,908,267,229
Number of sentences	95,119,665,584
Size compressed (counts only)	24 GB
Number of unigrams	13,588,391
Number of bigrams	314,843,401
Number of trigrams	977,069,902
Number of fourgrams	1,313,818,354
Number of fivegrams	1,176,470,663

Zipf' Law (1935,1949):

The frequency of any word is inversely proportional to its rank in the frequency table.



Neural Networks

Intuition: N-Gram Issues

Training:

the arizona corporations commission authorized

Test:

the colorado businesses organization ___

- Does this training example help here?
  - Not really. No count overlap.
- Intuition: hope to learn that similar words act similarly.

Intuition: N-Gram Issues

Training:

the arizona corporations commission authorized

Test:

the colorado businesses organization ___

- Does this training example help here?
  - Not really. No count overlap.
- Intuition: hope to learn that similar words act similarly.

Intuition: N-Gram Issues

Training:

the arizona corporations commission authorized

Test:

the colorado businesses organization ___

- Does this training example help here?
  - Not really. No count overlap.
- Intuition: hope to learn that similar words act similarly.

Alternative Approach

Treat as multi-class prediction!

Let $\mathcal{V}$ be all possible words in the language (English 10,000 - 100,000).
Predict over words.

Important ideas that you have seen .

1. **Neural Network** to learn feature representation.
2. **Softmax** for multiclass prediction of next word (out of $\mathcal{V}$)
3. **Cross-entropy** loss as training objective.

Language Modeling as Supervised Learning

THE ROOM WAS NOT VERY LIGHT A SMALL OBLONG READING LAMP ON THE DESK SHED GLOW ON POLISHED ___

- Training data is pairs are $(w_t, \{w_{t-3}, w_{t-2}, w_{t-1}\})$.
- Input is sentence up until the blank, output is next word prediction.
- Challenging multi-class prediction problem, feature representation matters.

Neural Network

$$p(w_t | \{w_{t-3}, w_{t-2}, w_{t-1}\}) = \sigma(z)_{w_t}$$

where

$$z = NN(\{w_{t-3}, w_{t-2}, w_{t-1}\})$$

Language Modeling as Supervised Learning

THE ROOM WAS NOT VERY LIGHT A SMALL OBLONG READING LAMP ON THE DESK SHED GLOW ON POLISHED ___

- Training data is pairs are $(w_t, \{w_{t-3}, w_{t-2}, w_{t-1}\})$.
- Input is sentence up until the blank, output is next word prediction.
- Challenging multi-class prediction problem, feature representation matters.

  Neural Network

$$p(w_t|\{w_{t-3}, w_{t-2}, w_{t-1}\}) = \sigma(z)_{w_t}$$

where

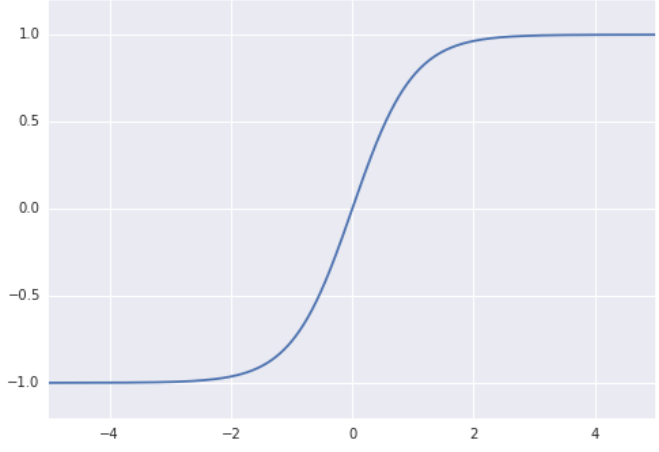
$$z = NN(\{w_{t-3}, w_{t-2}, w_{t-1}\})$$

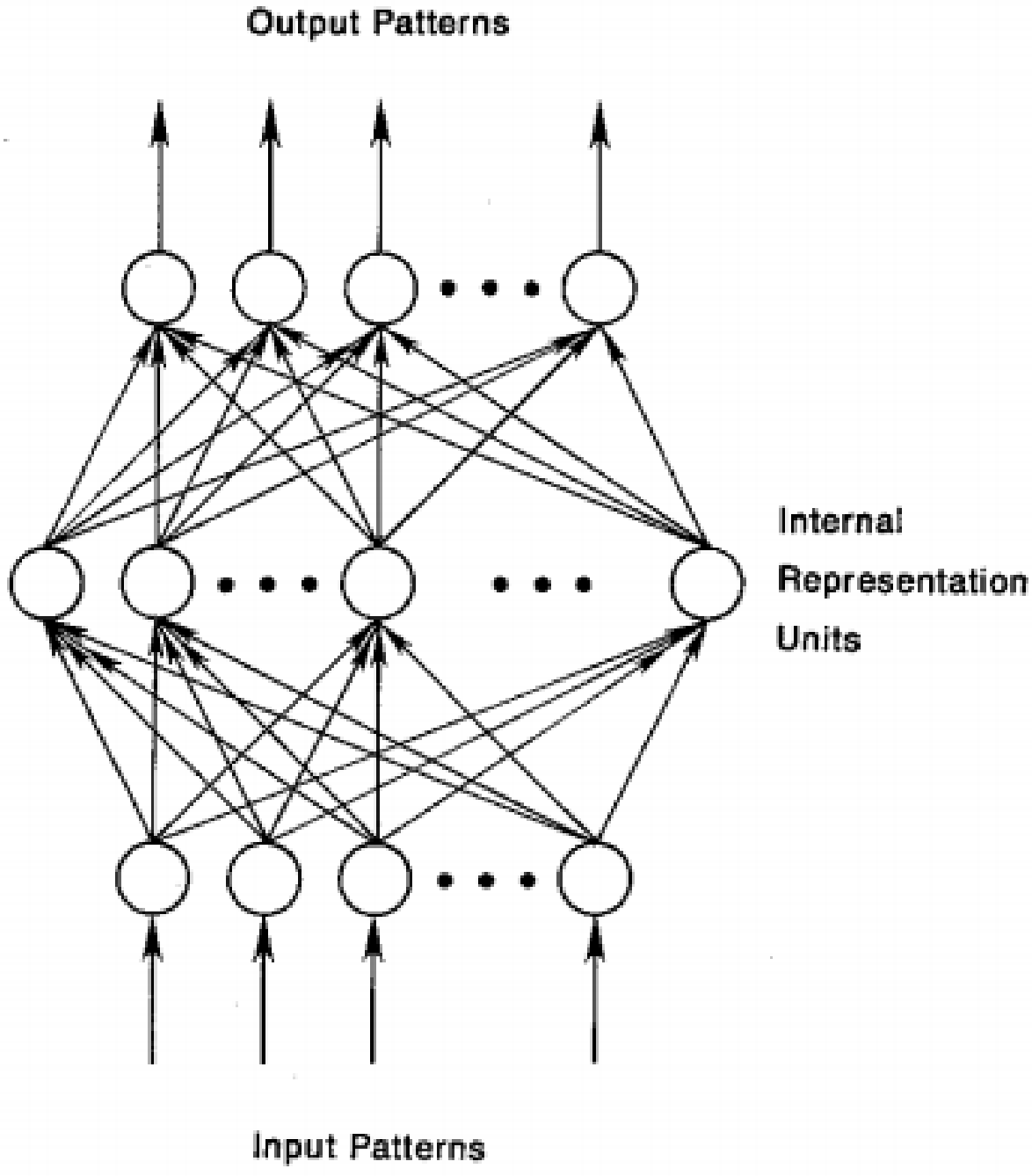
A neural network is a function approximator

- $NN(\mathbf{x}; \theta)$; a learned function from $\mathbf{x}$ with parameters θ.

$$NN_{MLP1}(\mathbf{x}) = \mathbf{W}^2 \tanh(\mathbf{W}^1 \mathbf{x} + \mathbf{b}^1) + \mathbf{b}^2$$

- $\mathbf{x}$; input
- $\mathbf{W}, \mathbf{b}$; parameters
- $\tanh$ is *non-linearity*

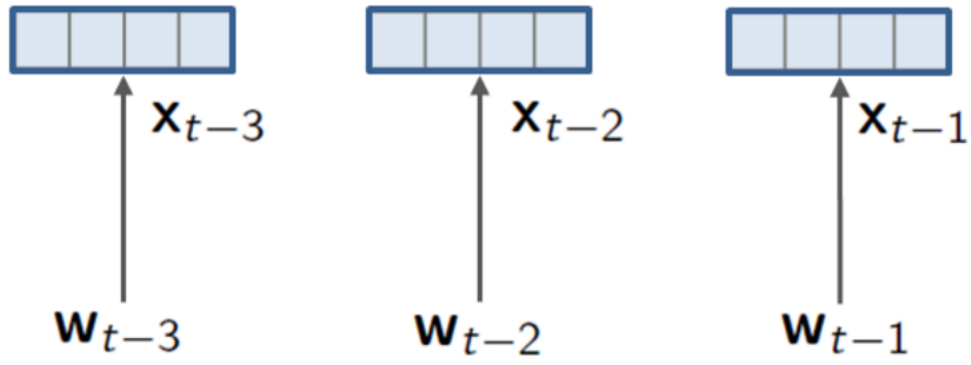



Output Patterns
Internal
Representation
Units
Input Patterns

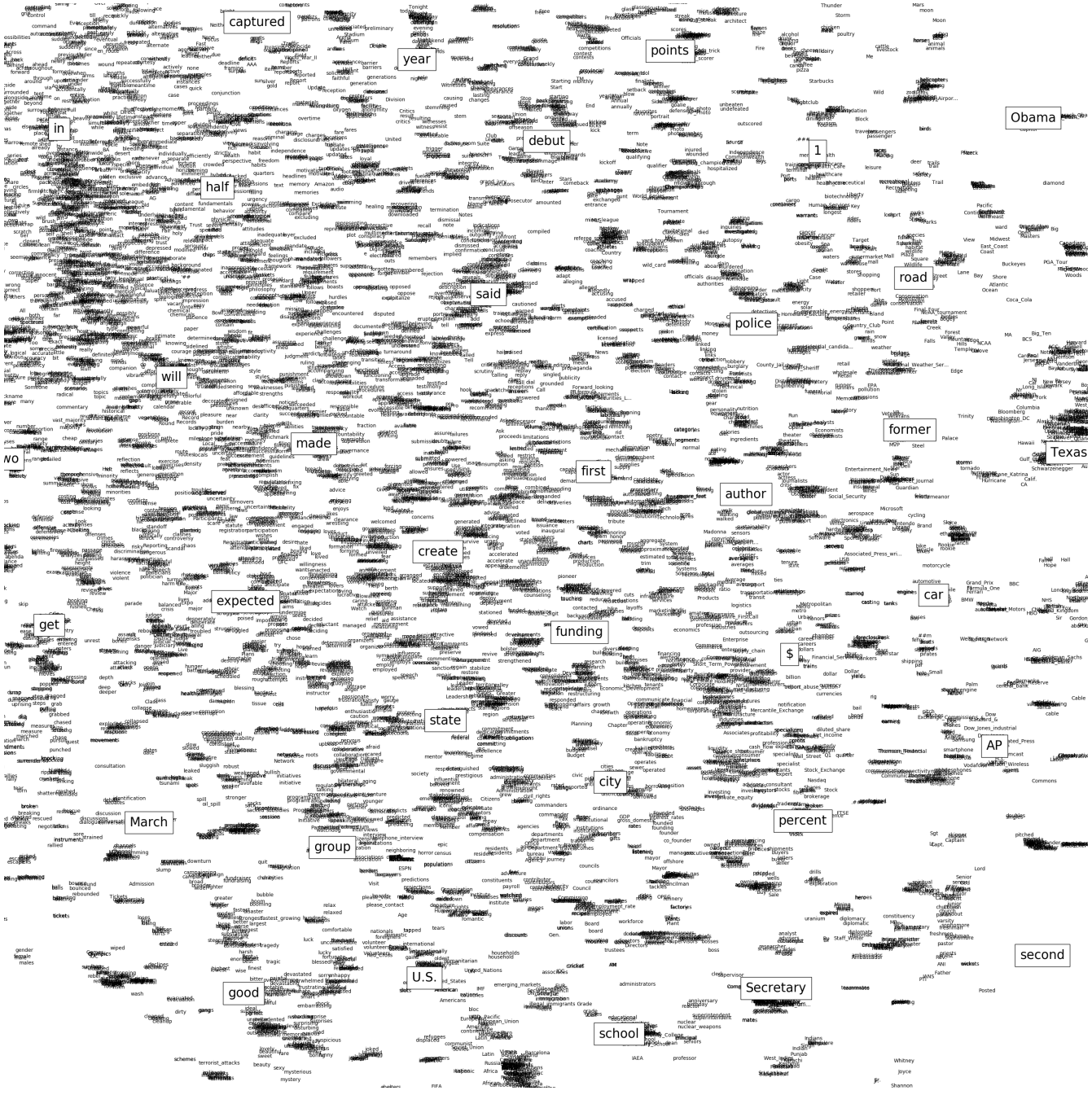
Word Representation

- How can a word be fed into a neural network?
- Each word has an associated numerical index, not like images or sounds.
- Need a way to know if two words are “close” to each other for the network.

Words Embeddings



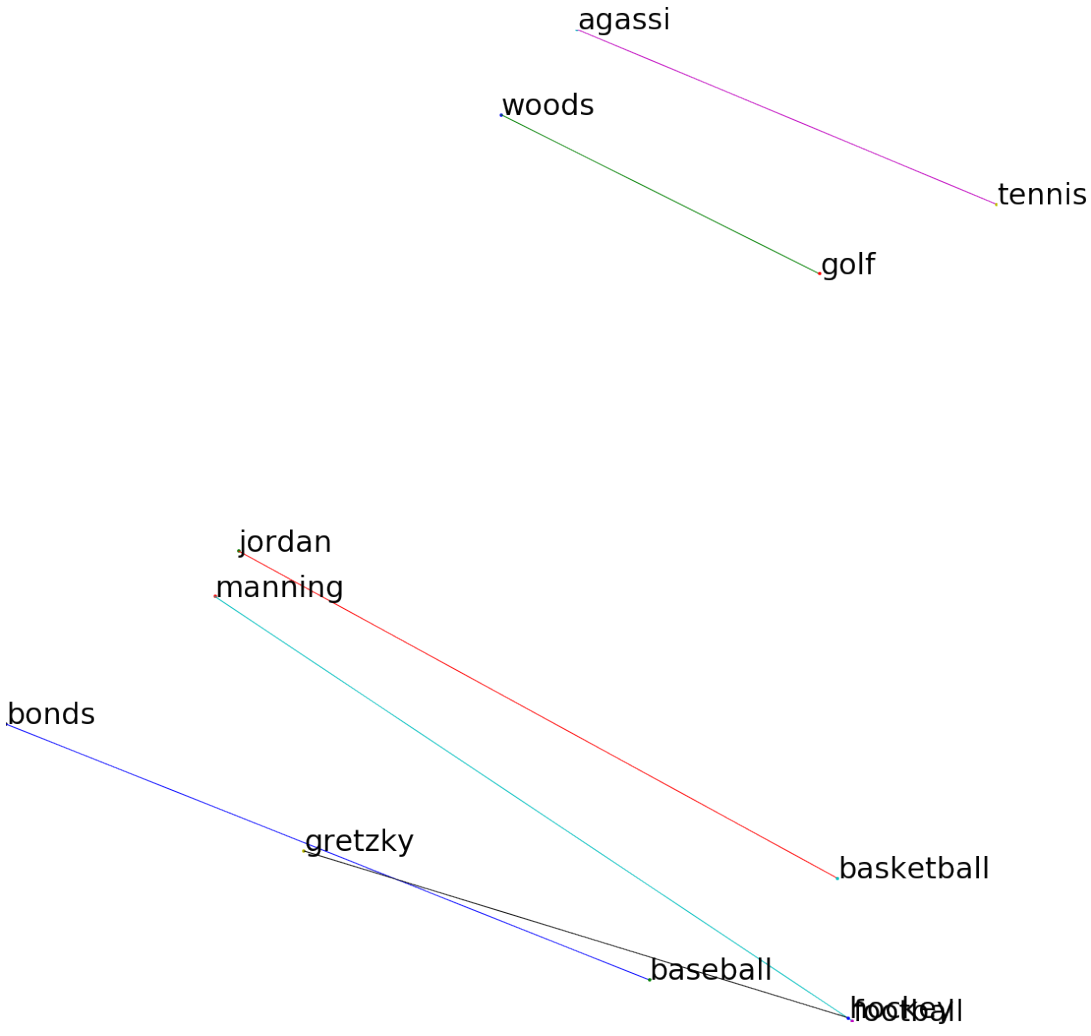
- Associate each words with an *embedding* vector, e.g. in 50 dimensions.
- Store this in a big table.
- "Move" the vectors around based on backpropagation, i.e. if the model makes the wrong prediction the vector associated with a word is updated.



[Words Vectors]

Notable Properties

After applying PCA to word embeddings

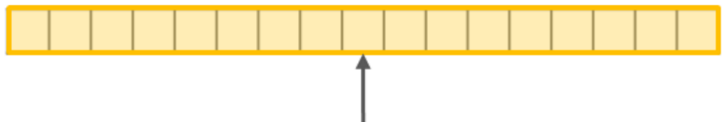


Modern Word Vectors

[Demo: TensorBoard]

Language Modeling with Neural Networks

Recall the definition of the **softmax**, with z the representation of the context.



$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

For language modeling K is huge! Need to compute for the full vocabulary.

Became possible to do with refinement of GPUs.

Problems With MLP Language Models

MLP language models provide a way to learn representations of words.

- Can be trained as a standard neural network.
- Allows us to learn words are similar.

Still have several issues.

- Have to differentiate between locations (2 words back, 5 words back).
- Only have a fixed amount of history.

Recurrent Neural Networks Models

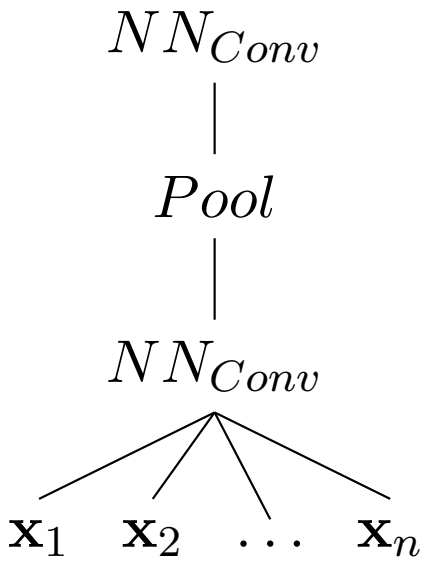
Recurrent Neural Networks

Main Idea: Build neural networks that are deep *in time.*

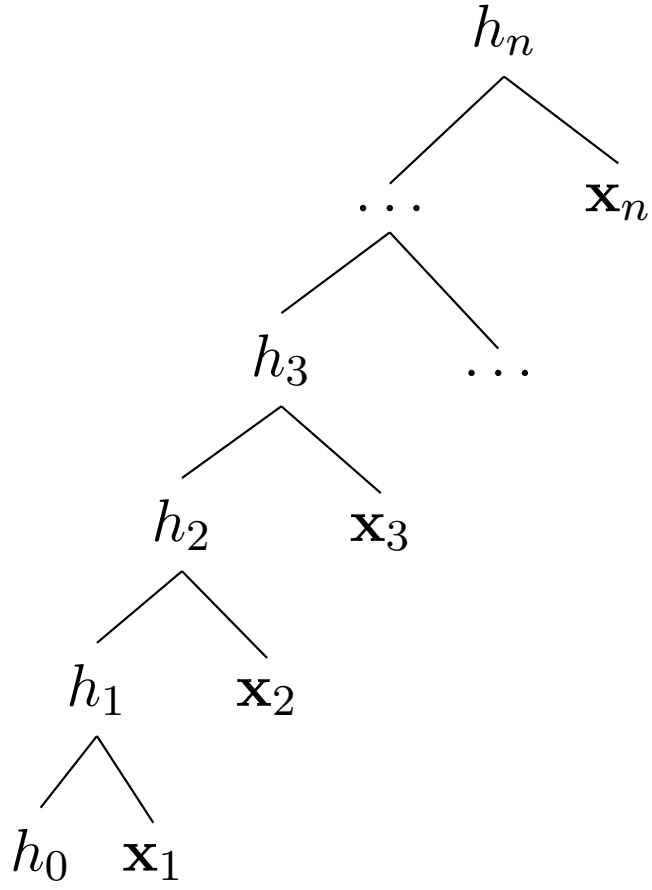
- Deep MLP/CNN: Stack multiple non-linear network units on-top of input. Learn richer feature representations.
- RNN: Use a non-linear transformations based on each input in a time-series.

RNN versus Deep Convolution

Convolution



RNN



RNN Mathematically

Let $h_0 \leftarrow 0$,

$$h_1 \leftarrow \textbf{RNN}(h_0, \mathbf{x}_1; \theta)$$

$$h_2 \leftarrow \textbf{RNN}(h_1, \mathbf{x}_2; \theta)$$

$$h_3 \leftarrow \textbf{RNN}(h_2, \mathbf{x}_3; \theta)$$

$$\vdots$$

$$h_n \leftarrow \textbf{RNN}(h_{n-1}, \mathbf{x}_n; \theta)$$

Where RNN is a neural network layer with the same weights θ applied at each time step, for instance:

$$\textbf{RNN}(h, \mathbf{x}) = \tanh(\mathbf{W}^a h + \mathbf{W}^b x)$$

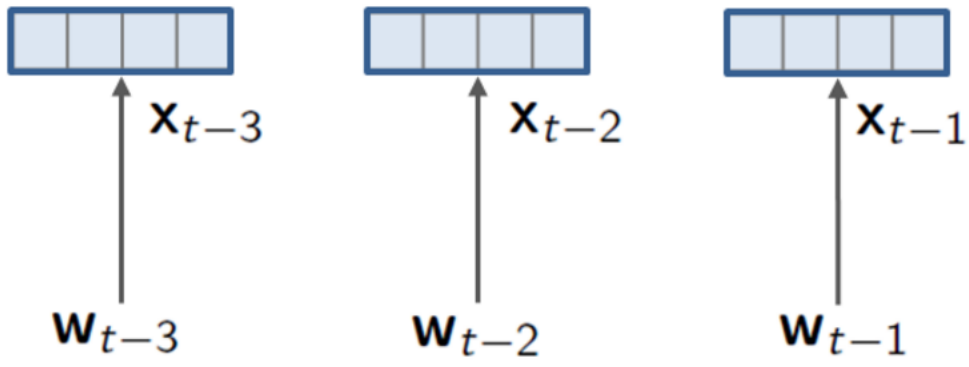
Why RNN?

- Get a single hidden vector for each time step.
- Vector can learn to capture important properties of the previous inputs.
- May allow us to make prediction based on history.

RNN For Language Modeling

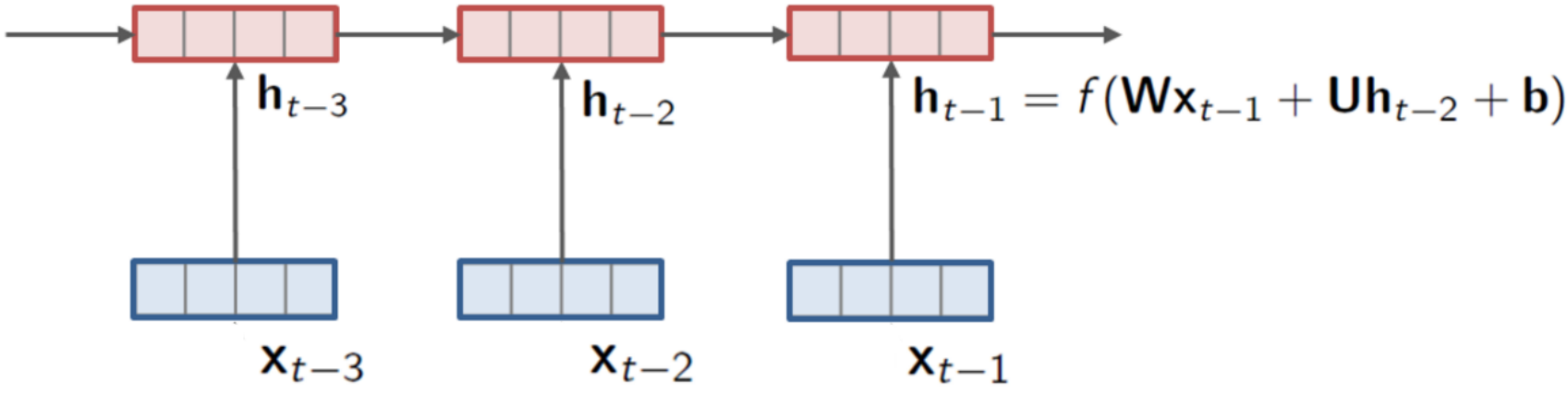
Embeddings	words	$\Rightarrow$	word vectors
RNNs	word vectors	$\Rightarrow$	hidden states
Softmax	hidden states	$\Rightarrow$	word prediction

Words Embeddings



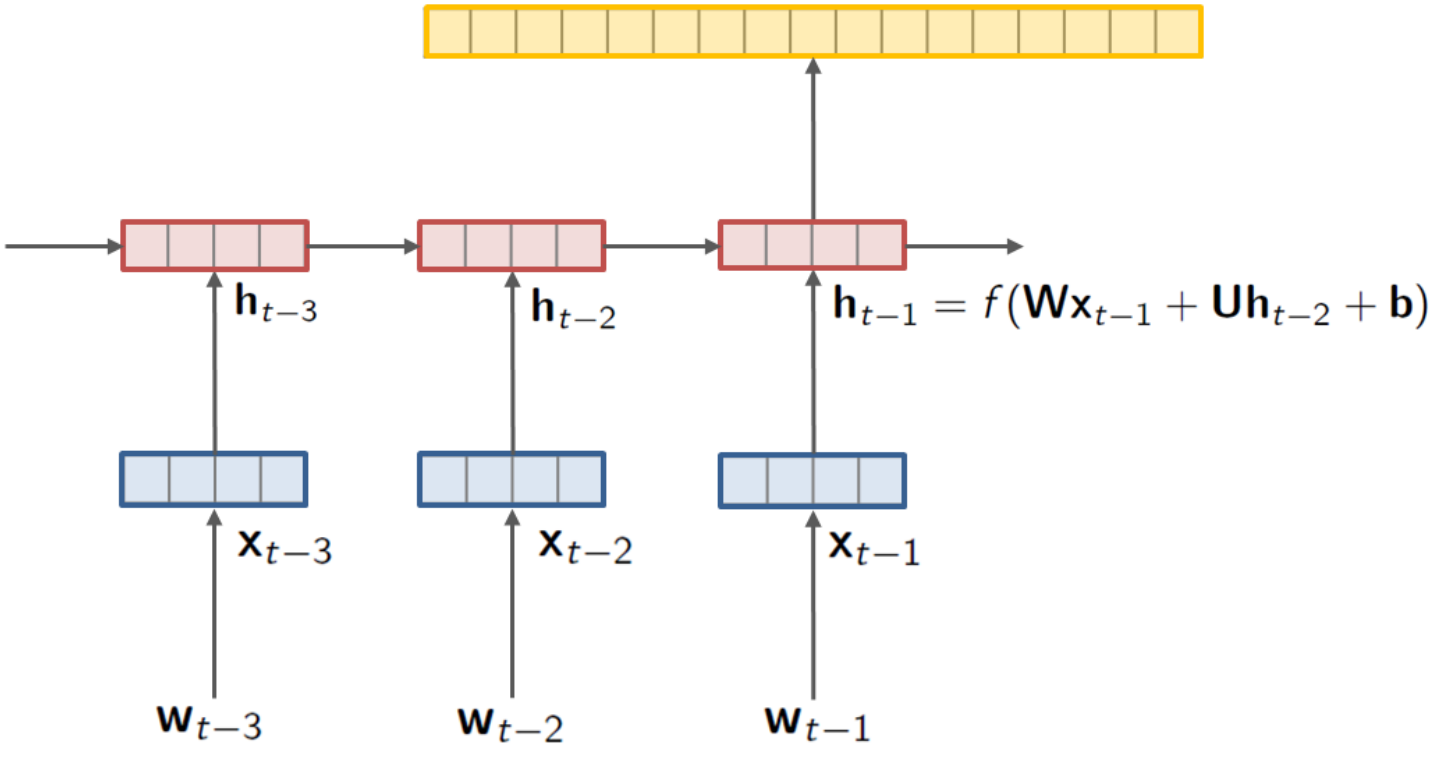
- Map words to vector space, just as before.

RNN



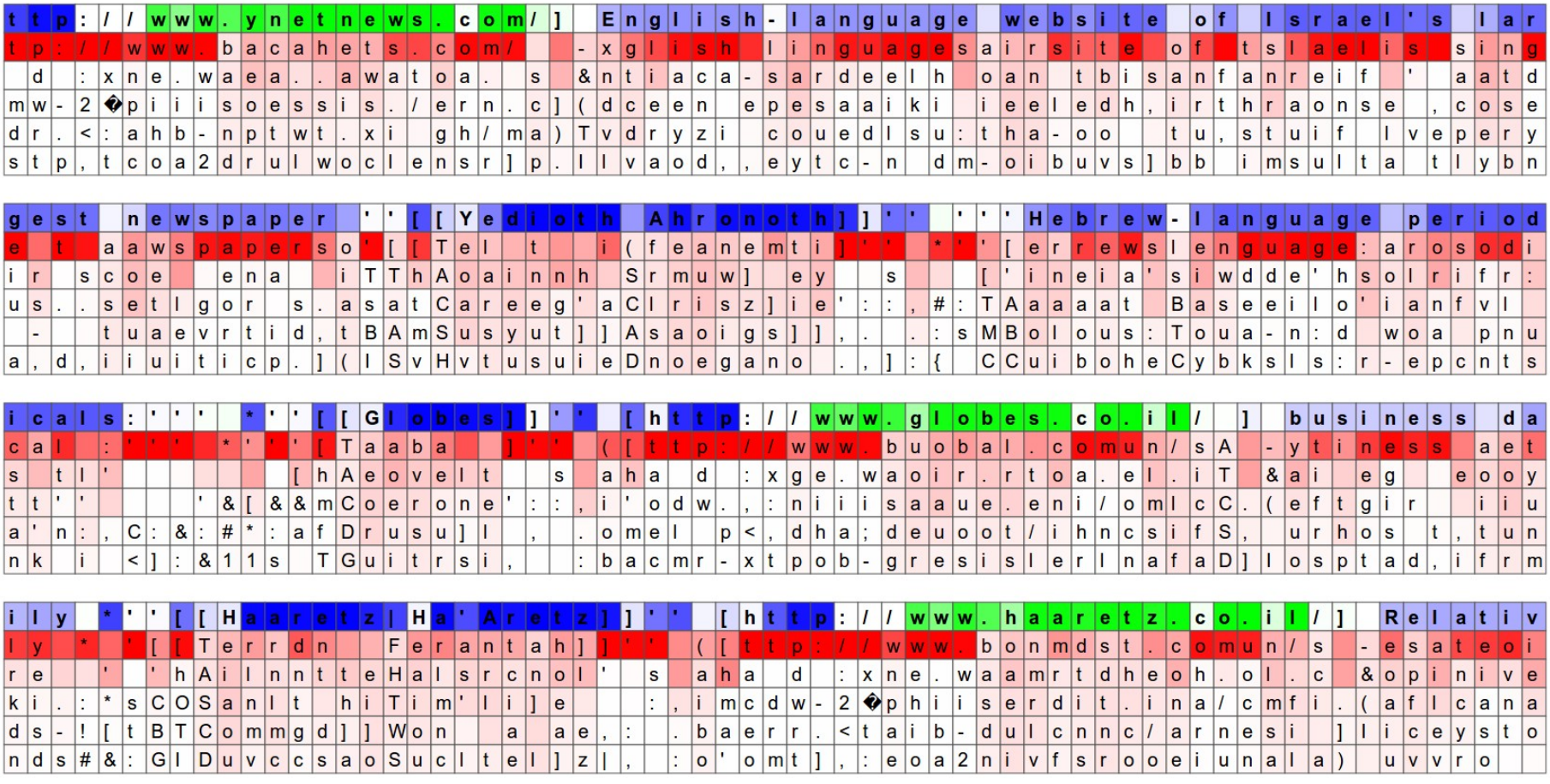
- Apply recurrent neural network over vector space of words to create hidden states.

Softmax



- Compute softmax over all possible next words to predict.

$$p(w_t \mid w_1, \ldots, w_{t-1}) = \sigma(\mathbf{W}\mathbf{h}_{t-1} + \mathbf{b})_w$$



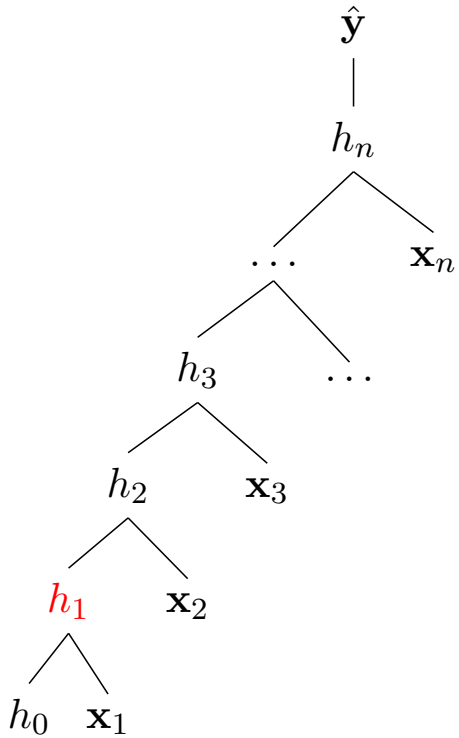
(Karpathy et al, 2015)

How do we learn the model?

- RNNs are trained with SGD and Backprop (surprise)
- Implementation can be complicated, mainly for efficiency.
- Called *backpropagation through time* (BPTT).

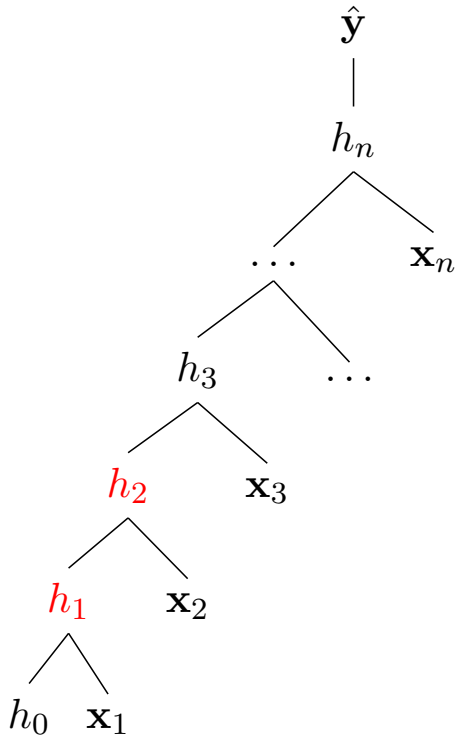
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



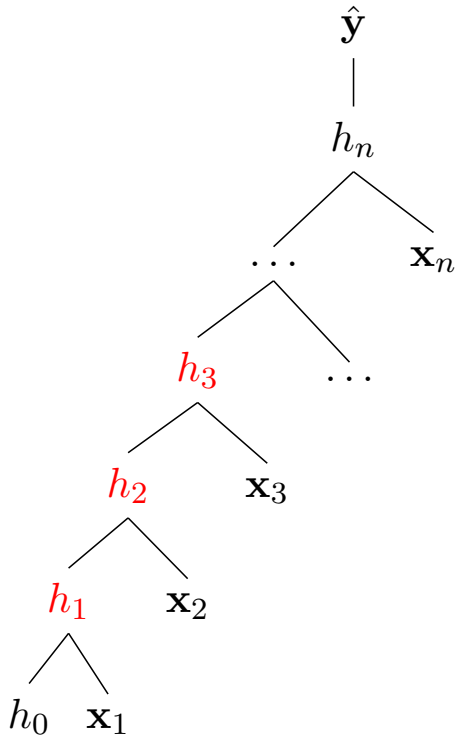
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



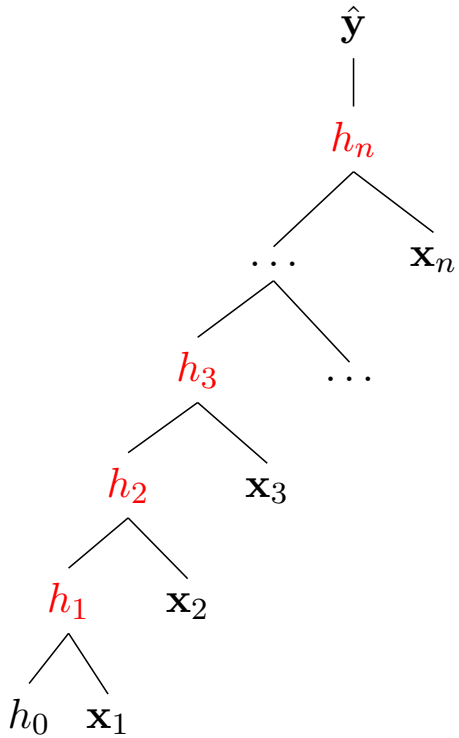
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



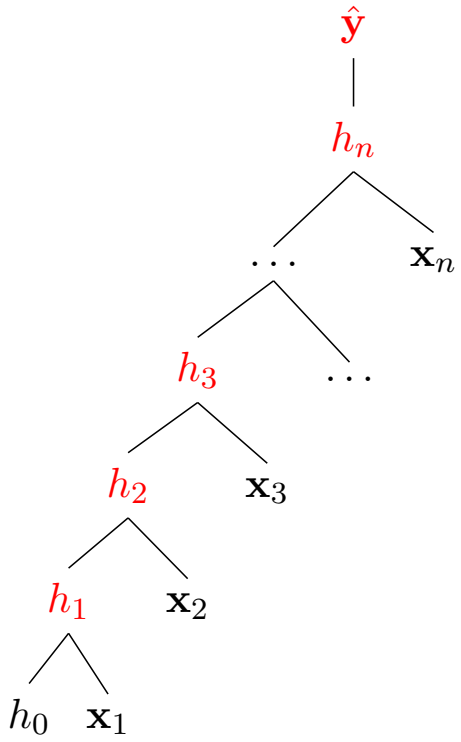
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



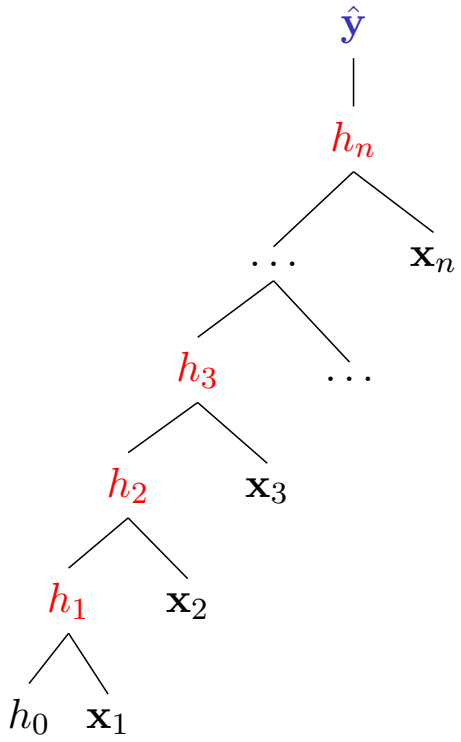
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



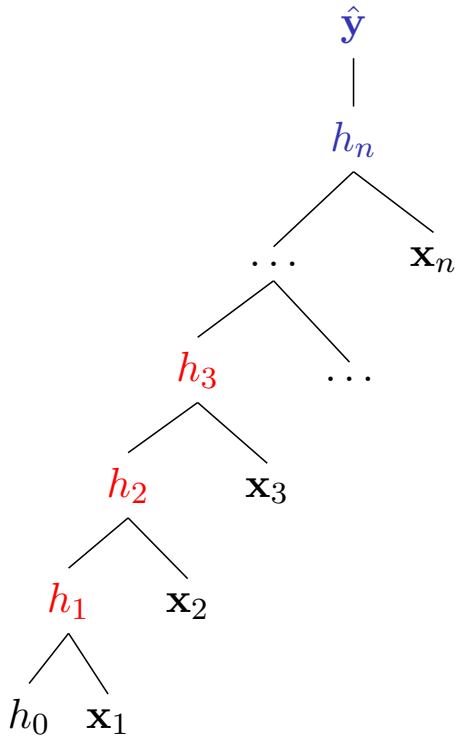
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



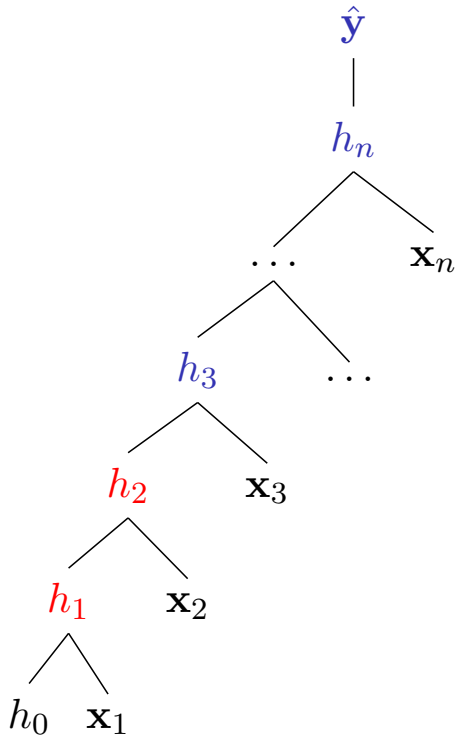
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



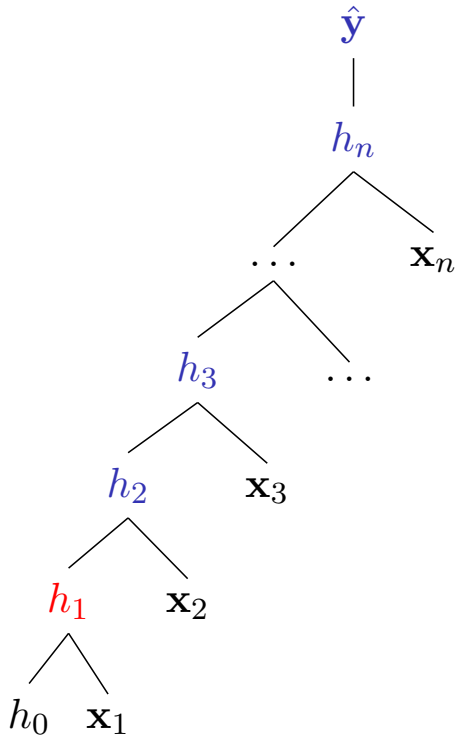
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



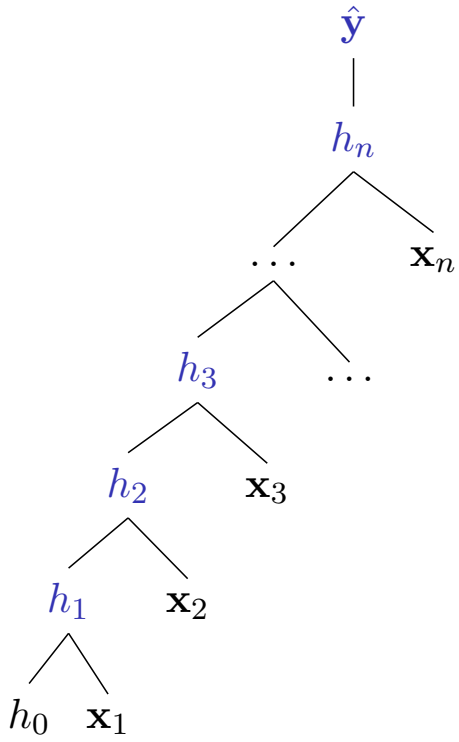
BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



BPTT (One Word)

- Run forward propagation.
- Run backward propagation.
- Update all weights (shared)



Issues

- Can be inefficient, but batch/GPUs help.
- Model is much deeper than previous approaches.
  - This matters a lot, focus of next class.
- Variable-size model for each sentence.
  - Have to be a bit more clever in Keras.

Application of RNNs

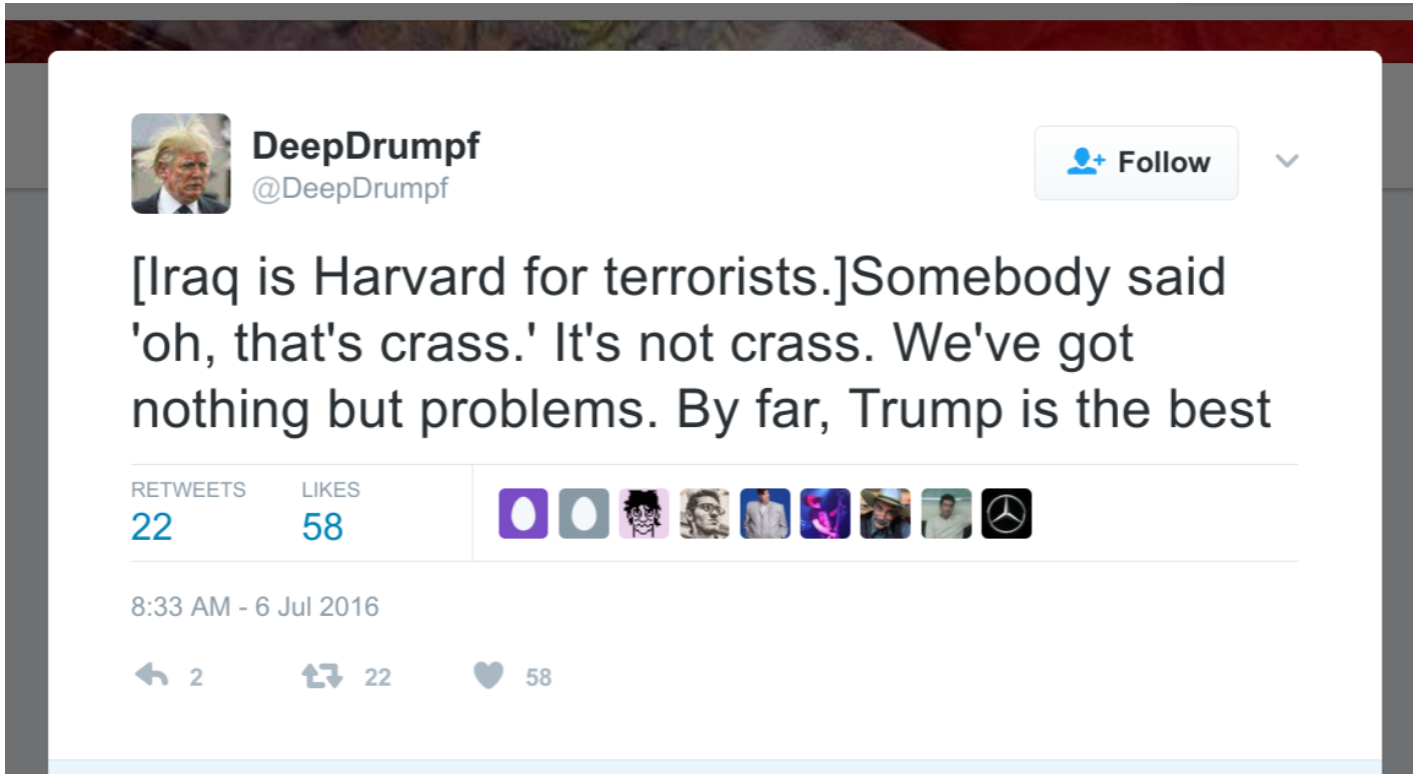
RNN models have lead to a **major** increase in the accuracy of language models.

Why did this matter?

Application of RNNs

RNN models have lead to a **major** increase in the accuracy of language models.

Why did this matter?



Language Modeling Applications

- Speech Recognition
- Machine Translation
- Summarization
- Dialogue
- Soft Keyboards
- Word Correction
- Text Simplification
- ...

Next Class

- Easier to train variants of RNN (Long-Short Term Memory)
- LSTMs for Machine Translation
- Many Cool Applications...

Thanks!

References I