Sequence Models

Jordan Boyd-Graber

University of Maryland

RNNs

Slides adapted from Richard Socher and Phillip Koehn

Language models

- **Language models** answer the question: How likely is a string of English words good English?
- Autocomplete on phones and websearch
- Creating English-looking documents
- Very common in machine translation systems
  - Help with reordering / style

$$p_{\text{lm}}(\text{the house is small}) > p_{\text{lm}}(\text{small the is house})$$

  - Help with word choice

$$p_{\text{lm}}(\text{I am going home}) > p_{\text{lm}}(\text{I am going house})$$

Fill in the blank

I have a sad story to tell you
It may hurt your feelings a bit
Last night when I walked into my bathroom
I stepped in a big pile of . . .

Fill in the blank

I have a sad story to tell you
It may hurt your feelings a bit
Last night when I walked into my bathroom
I stepped in a big pile of . . .

Language Modeling: The Good Old Days

n-gram models

- Have a big corpus
- Count n-gram sequences
- Estimate $p(w_n \mid w_{n-1} \dots w_{n-k}) =$

$$\frac{\text{Count}(w_{n-k} \dots w_{n-1} w_n)}{\text{Count}(w_{n-k} \dots w_{n-1})} \tag{1}$$

Language Modeling: The Good Old Days

n-gram models

- Have a big corpus
- Count n-gram sequences
- Estimate $p(w_n \mid w_{n-1} \dots w_{n-k}) =$

$$\frac{\text{Count}(w_{n-k} \dots w_{n-1} w_n)}{\text{Count}(w_{n-k} \dots w_{n-1})} \tag{1}$$

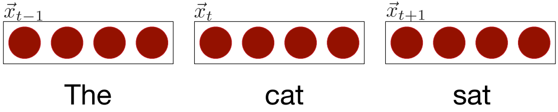
Log-linear models

- Define a feature vector f based on word w and context c
- (Can include n-gram features)
- Learn β from data
- Then $p(w \mid c) =$

$$\frac{\exp\{\beta f(w, c)\}}{\sum_v \exp\{\beta f(v, c)\}} \tag{2}$$

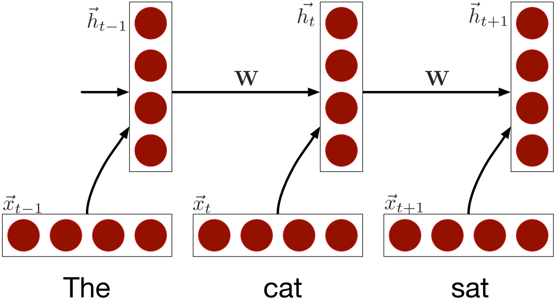
Reccurent Neural Networks

Use Word2Vec or learn representations from scratch



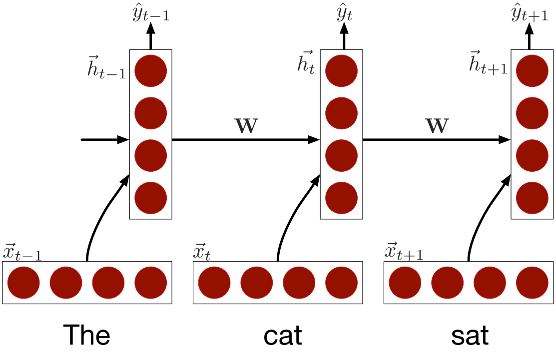
Reccurent Neural Networks

Each hidden state has $D = 500$ or so



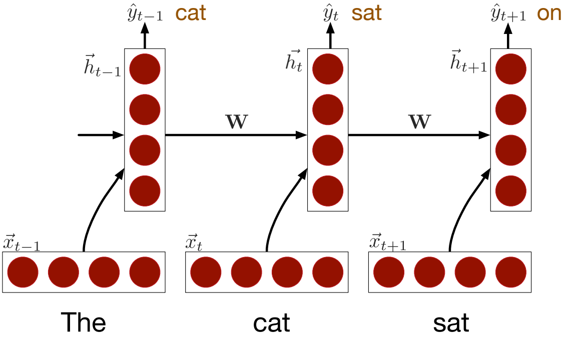
Reccurent Neural Networks

Transform hidden state to V (vocab) matrix $\mathbf{W}^{(s)}\vec{h}_t$



Reccurent Neural Networks

Take softmax to get real distribution



RNN parameters

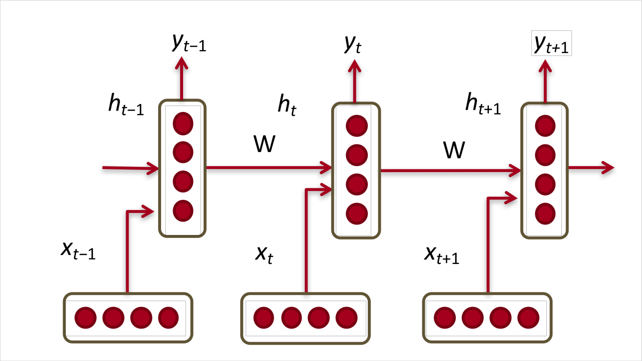
$$h_t = f(W^{(hh)} h_{t-1} + W^{(hx)} x_t) \quad (3)$$

$$\hat{y}_t = \text{softmax}(W^{(S)} h_t) \quad (4)$$

$$P(x_{t+1} = v_j \mid x_t, \ldots x_1) = \hat{y}_{t,j} \quad (5)$$

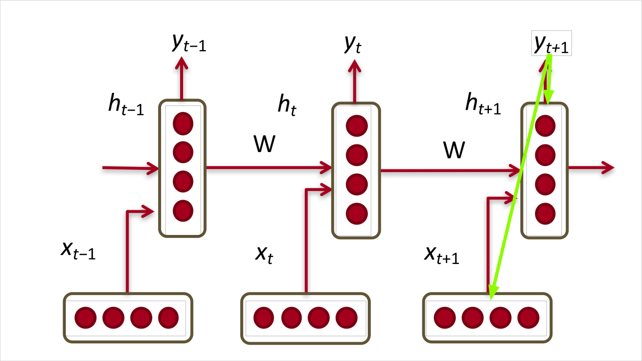
- Learn parameter h_0 to initialize hidden layer
- x_t is representation of input (e.g., word embedding)
- $\hat{y}$ is probability distribution over vocabulary

Training Woes



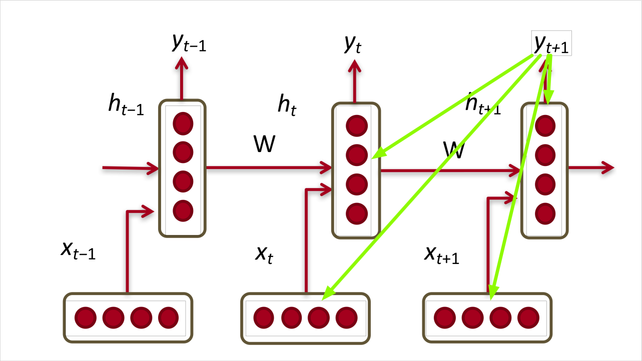
Multiplying same matrix over and over

Training Woes



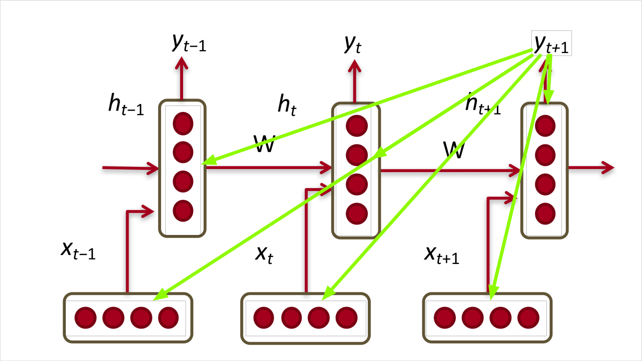
Multiplying same matrix over and over

Training Woes



Multiplying same matrix over and over

Training Woes



Multiplying same matrix over and over

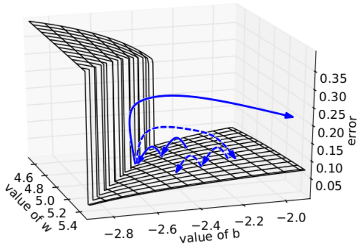
Vanishing / Exploding Gradient

- Work out the math:
  - Define β_W / β_h as upper bound of norms of W, h
  - Bengio et al 1994: Partial derivative is $(\beta_W \beta_h)^{t-k}$
  - This can be very small or very big
- If it's big, SGD jumps too far
- If it's small, we don't learn what we need: "Jane walked into the room with John, who wasn't paying attention to what was going on. After poking him to get his attention, John said hi to ____"

Gradient Clipping

Algorithm 1 Pseudo-code for norm clipping the gradients whenever they explode

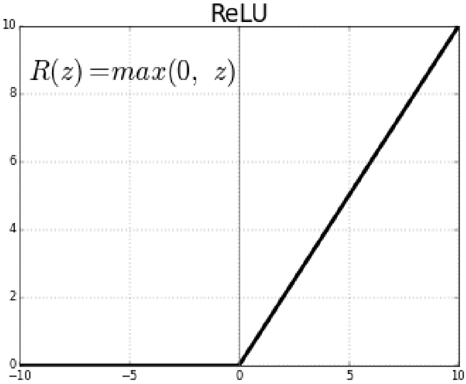
$\hat{\mathbf{g}} \leftarrow \frac{\partial \mathcal{E}}{\partial \theta}$
if $\|\hat{\mathbf{g}}\| \geq threshold$ **then**
  $\hat{\mathbf{g}} \leftarrow \frac{threshold}{\|\hat{\mathbf{g}}\|}\hat{\mathbf{g}}$
end if



From Pascanu et al. 2013

- If they get too big, stop at boundary
- Prevents (dashed) values from jumping around (solid)

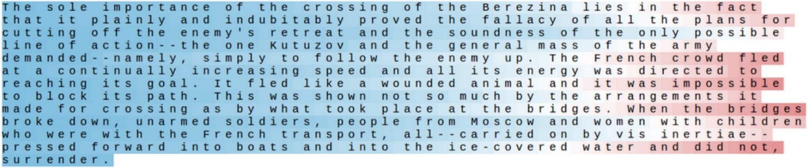
Fixing Vanishing Gradients



- ReLU activation
- Initialize W to identity matrix

Vizualization from Karpathy et al

Cell sensitive to position in line:



Vizualization from Karpathy et al

Cell that turns on inside quotes:

"You mean to imply that I have nothing to eat out of.... On the contrary, I can supply you with everything even if you want to give dinner parties," warmly replied Chichagov, who tried by every word he spoke to prove his own rectitude and therefore imagined Kutuzov to be animated by the same desire.

Kutuzov, shrugging his shoulders, replied with his subtle penetrating smile: "I meant merely to say what I said."

Vizualization from Karpathy et al

Cell that robustly activates inside if statements:

```
static int __dequeue_signal(struct sigpending *pending, sigset_t *mask,
   siginfo_t *info)
{
 int sig = next_signal(pending, mask);
 if (sig) {
  if (current->notifier) {
   if (sigismember(current->notifier_mask, sig)) {
    if (!(current->notifier)(current->notifier_data)) {
     clear_thread_flag(TIF_SIGPENDING);
     return 0;
    }
   }
  }
  collect_signal(sig, pending, info);
 }
 return sig;
}
```

RNN Recap

- Simple model
- Complicated training (but good toolkits available)
- Do we need to remember everything?