Natural Language Processing

Anoop Sarkar
anoopsarkar.github.io/nlp-class

Simon Fraser University

October 25, 2018

Natural Language Processing

Anoop Sarkar
anoopsarkar.github.io/nlp-class

Simon Fraser University

Part 1: Neural Language Models

Long distance dependencies

Example

- He doesn't have very much confidence in himself
- She doesn't have very much confidence in herself

n-gram Language Models: $P(w_i \mid w_{i-n+1}^{i-1})$

$$P(\text{himself} \mid \text{confidence, in})$$
$$P(\text{herself} \mid \text{confidence, in})$$

What we want: $P(w_i \mid w_{<i})$

$$P(\text{himself} \mid \text{confidence}, \ldots, \text{him})$$
$$P(\text{herself} \mid \text{confidence}, \ldots, \text{her})$$

Long distance dependencies

Other examples

- **Selectional preferences**: *I ate lunch with a fork* vs. *I ate lunch with a backpack*
- **Topic**: *Babe Ruth was able to touch the home plate yet again* vs. *Lucy was able to touch the home audiences with her humour*
- **Register**: Consistency of register in the entire sentence, e.g. informal (Twitter) vs. formal (scientific articles)

Language Models

Chain Rule and ignore some history: the trigram model

$$
\begin{aligned}
& p(w_1, \ldots, w_n) \\
& \quad \approx \quad p(w_1)p(w_2 \mid w_1)p(w_3 \mid w_1, w_2) \ldots p(w_n \mid w_{n-2}, w_{n-1}) \\
& \quad \approx \quad \prod_t p(w_{t+1} \mid w_{t-1}, w_t)
\end{aligned}
$$

How can we address the long-distance issues?

- Skip n-gram models. Skip an arbitrary distance for n-gram context.
- Variable n in n-gram models that is adaptive
- **Problems**: Still "all or nothing". Categorical rather than soft.

Neural Language Models

Use Chain rule and approximate using a neural network

$$p(w_1, \ldots, w_n) \approx \prod_t p(w_{t+1} \mid \underbrace{\phi(w_1, \ldots, w_t)}_{\text{capture history with vector } s(t)})$$

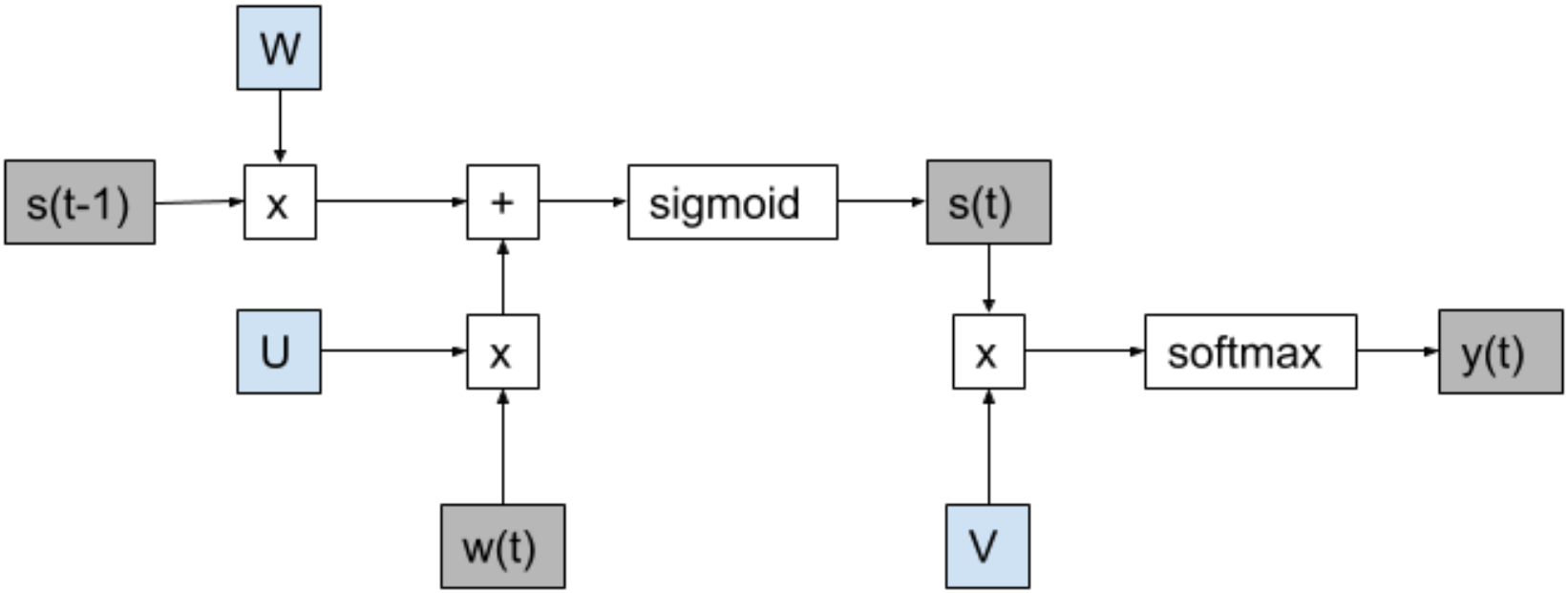
Recurrent Neural Network

- Let y be the output w_{t+1} for current word w_t and history $w_1, \ldots, w_t$
- $s(t) = f(U_{xh} \cdot w(t) + W_{hh} \cdot s(t-1))$ where f is sigmoid
- $s(t)$ encapsulates history using single vector of size h
- Output word at time step w_{t+1} is provided by $y(t)$
- $y(t) = g(V_{hs} s(t))$ where g is softmax

Neural Language Models

Recurrent Neural Network

Computational Graph for an RNN Language Model



Acknowledgements

Many slides borrowed or inspired from lecture notes by Michael Collins, Chris Dyer, Kevin Knight, Philipp Koehn, Adam Lopez, Graham Neubig and Luke Zettlemoyer from their NLP course materials.

All mistakes are my own.

A big thank you to all the students who read through these notes and helped me improve them.