Recurrent Neural Networks

Time Series
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
    - chemical plants, data centers?
Standard HMM

◆ HMM learning problem: Estimate $T$ and $O$

- Optimization done via EM
  - Or spectral methods
- History is forgotten with an exponential decay
Simple Recurrent Neural Net

\[ s_t = \tanh(U x_t + W s_{t-1}) \]
\[ o_t = \text{softmax}(V s_t) \]

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

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

\[ \sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for } j = 1, \ldots, K. \]
Like HMMs, unroll RNNs in time

\( x_t = \text{input (e.g. a word)} \)

\( s_t = \text{hidden state} \)

\( o_t = \text{output (e.g. probability of the next word)} \)

http://www.nature.com/nature/journal/v521/n7553/full/nature14539.html
Neural nets can take actions

emission

action (A)
Recurrent Neural Nets (RNNs)

\[ s_t = \tanh(U x_t + W s_{t-1}) \]
\[ o_t = \text{softmax}(V s_t) \]

Can use multiple layers
Gated RNNs

- Standard RNNs, like HMMs, tend to forget things exponentially fast
- Solution: Gated RNN
  - Stores hidden state
    \[ z = \sigma(U^z x_t + W^z s_{t-1}) \]  
    \[ r = \sigma(U^r x_t + W^r s_{t-1}) \]  
    \[ h = \tanh(U^h x_t + W^h (s_{t-1} \circ r)) \]  
    \[ s_t = (1-z) \circ h + z \circ s_{t-1} \]  
  - \( z \): update gate  
  - \( r \): reset gate
  - \( o_t \): prediction
  - \( x_t \): input
  - \( s_t \): hidden state

- \( r=0 \) resets \( h \)
- \( z=1 \) keeps state
- \( z=0 \) updates it to \( h \)
- \( r=1 \)'s, \( z=0 \)'s gives simple RNN

\( \circ \) is pointwise multiplication

http://deeplearning.net/tutorial/lstm.html
Recurrent Nets can be stacked
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

\[
\begin{align*}
  x_t & \quad \text{observation} \\
  s_t & \quad \text{hidden state} \\
  o & = \text{output gate (not output!)}
\end{align*}
\]

\[
\begin{align*}
  i &= \sigma(U^i x_t + W^i s_{t-1}) \\
  f &= \sigma(U^f x_t + W^f s_{t-1}) \\
  o &= \sigma(U^o x_t + W^o s_{t-1}) \\
  h &= \tanh(U^h x_t + W^h s_{t-1}) \\
  c_t &= f \circ c_{t-1} + i \circ h \\
  s_t &= o \circ \tanh(c_{t-1})
\end{align*}
\]

http://deeplearning.net/tutorial/lstm.html
Recurrent Neural Nets

- Predict the next observation given past observations
- Or 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

The lazy brown fox jumped over the ...

Jeff Dean, google
https://www.youtube.com/watch?v=90-S1M7Ny_o&spfreload=1
Encode and Decode = translate

The quick brown fox jumped over the ...

Le renard brun rapide saute par dessus ...

Jeff Dean, google
... or a chatbot
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.
Can map images to words

Connect sequence and image models, you get a captioning system

“A close up of a child holding a stuffed animal”
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.
Language inputs to RNN

- Words ("one-hot")
- Characters ("one-hot")
- Word embeddings
  - Typically 300 dimensional
Alternative RNN models

- External memory
- Attentional models

![Diagram of Metamind model]

Metamind “Ask Me Anything:”
Attention via gating

\[ G(\text{candidate fact } c = s_j, \text{ previous state } m, \text{ question } q) \]

\[ g_t^i = G(c_t, m_{i-1}, q) \]

\[ e_t^i = g_t^i \text{ GRNN}(c_t, e_{t-1}^i) + (1-g_t^i) e_{t-1}^i \]
Dynamic Network Summary

- **Gated Neural Nets generalize HMMs, Kalman filters**
  - But are far more powerful!
- **They have replaced HMMs for speech to text and machine translation**
- **Lots of black magic “engineering”**
  - Unclear what matters about the network structure
    - Number and size of layers, regularization
    - Forms of gating (LSTM …), attention …
  - Gradient descent is tricky
- **Good software: tensorflow, theano …**