Neural Network with Memory

Hung-yi Lee
Memory is important

Input:
2 dimensions

\[
\begin{align*}
\mathbf{x}^1 &= 4 \quad 4 \quad 1 \\
\mathbf{x}^2 &= 7 \quad 7 \quad 1 \\
\mathbf{x}^3 &= 1 \quad 4 \quad 4
\end{align*}
\]

Output:
1 dimension

\[
\begin{align*}
\hat{\mathbf{y}}^1 &= 1 \\
\hat{\mathbf{y}}^2 &= 2 \\
\hat{\mathbf{y}}^3 &= 3
\end{align*}
\]

\[
\begin{align*}
1 &+ 1 \quad 7 \quad 7 \\
\hline
3 &2 &1
\end{align*}
\]

Network needs memory to achieve this
Memory is important

Network with Memory

\[
\begin{array}{c}
\hat{y}^1 \\
\hat{y}^2 \\
\hat{y}^3 \\
\end{array}
\begin{array}{c}
1 \\
2 \\
3 \\
\end{array}
\]

\[
\begin{array}{c}
x^1 \\
x^2 \\
x^3 \\
\end{array}
\begin{array}{c}
4 \quad 7 \\
4 \quad 7 \\
1 \quad 1 \\
\end{array}
\]

\[
\begin{array}{c}
x_1 \\
x_2 \\
\end{array}
\begin{array}{c}
4 \\
7 \\
\end{array}
\]
Memory is important

Network with Memory

\[
\begin{align*}
\hat{y}^1 &= 1 \\
\hat{y}^2 &= 2 \\
\hat{y}^3 &= 3 \\
\end{align*}
\]

\[
\begin{align*}
x^1 &= 4 \quad 7 \\
x^2 &= 4 \quad 7 \\
x^3 &= 1 \quad 1 \\
\end{align*}
\]
Memory is important

Network with Memory

$$x^1 \quad x^2 \quad x^3$$

$$\hat{y}^1 \quad \hat{y}^2 \quad \hat{y}^3$$

$$4 \quad 4 \quad 1$$
$$7 \quad 7 \quad 1$$

$$y$$

$$c_1$$

$$x_1 \quad x_2$$

$${3 \quad \quad 3}$$

$${10 \quad 1}$$

$${-10 \quad 1}$$

$${0 \quad 0}$$

$${1 \quad 1}$$

$${1 \quad 1}$$
Outline

Vanilla Recurrent Neural Network (RNN)

Variants of RNN

Long Short-term Memory (LSTM)
Outline

Vanilla Recurrent Neural Network (RNN)

Variants of RNN

Long Short-term Memory (LSTM)
Application

- (Simplified) Speech Recognition

We use DNN. All the frames are considered independently.
RNN input: \[ x^1, x^2, x^3, \ldots, x^N \]

Input of RNN is one utterance

The order cannot change.

\[ y^1 = \text{softmax}(W^o a^1) \]

\[ a^1 = \sigma(W^i x^1 + W^h 0) \]

Memory: \[ a^1 \]

Copy: \[ W^o \]

\[ x^1 \]
RNN input:

\[ x^1 \ x^2 \ x^3 \ \ldots \ x^N \]

\[ y^1 \ y^2 \]

The order cannot change.

Input of RNN is one utterance

\[ y^2 = \text{softmax}(W^o a^2) \]

\[ a^2 = \sigma(W^i x^2 + W^h a^1) \]

\[ W^o \]

\[ W^i \]

\[ W^h \]

memory

\[ a^1 \ a^2 \]

The order cannot change.
The order cannot change.

Input of RNN is one utterance.

\[ y^3 = \text{softmax}(W^o a^3) \]

\[ a^3 = \sigma(W^i x^3 + W^h a^2) \]
RNN

Input data: \( x^1 \), \( x^2 \), \( x^3 \), ……, \( x^N \)

Output \( y^i \) depends on \( x^1 \), \( x^2 \), ……, \( x^i \)

The same network is used again and again.

Input of RNN is one utterance.
RNN

Input data: \( x^1, x^2, x^3, \ldots, x^N \)

Input of RNN is one utterance

The same network is used again and again.

Output \( y^i \) depends on \( x^1, x^2, \ldots, x^i \)
Cost

RNN input: \( x^1 \ x^2 \ x^3 \ \ldots \ x^N \)

RNN output: \( y^1 \ y^2 \ y^3 \ \ldots \ y^N \)

RNN output: \( \hat{y}^1 \ \hat{y}^2 \ \hat{y}^3 \ \ldots \ \hat{y}^N \)

\[
C = \frac{1}{2} \sum_{n=1}^{N} ||y^n - \hat{y}^n||^2
\]

\[
C = \frac{1}{2} \sum_{n=1}^{N} -\log y^n_{rn}
\]
Training

\[ \hat{y}^1 \]
\[ \downarrow \]
\[ y^1 \]

\[ \hat{y}^2 \]
\[ \downarrow \]
\[ y^2 \]

\[ \hat{y}^3 \]
\[ \downarrow \]
\[ y^3 \]

\[ x^1 \]

\[ x^2 \]

\[ x^3 \]

\( w \) is an element in \( W^h, W^i \) or \( W^o \) \( \rightarrow \) \( w \leftarrow w - \eta \partial C / \partial w \)

Backpropagation through time (BPTT)

RNN Training is very difficult in practice.
More Applications

• Input and output are vector sequences with **the same length**

\[ \begin{align*}
  y^1 & \quad y^2 & \quad y^3 & \quad y^4 \\
  PN & \quad V & \quad D & \quad N \\
  x^1 & \quad x^2 & \quad x^3 & \quad x^4 \\
  \uparrow & \quad \uparrow & \quad \uparrow & \quad \uparrow \\
  \end{align*} \]

John saw the saw.

**POS Tagging**
More Applications

• Name entity recognition
  • Identifying names of people, places, organizations, etc. from a sentence
  • Harry Potter is a student of Hogwarts and lived on Privet Drive.
    • people, organizations, places, not a name entity

• Information extraction
  • Extract pieces of information relevant to a specific application, e.g. flight booking
  • I would like to leave Boston on November 2nd and arrive in Taipei before 2 p.m.
    • place of departure, destination, time of departure, time of arrival, other
Outline

Vanilla Recurrent Neural Network (RNN)

Variants of RNN

Long Short-term Memory (LSTM)
Elman Network & Jordan Network

**Elman Network**

\[ y_t \]

\[ x_t \]

\[ \mathbf{W}^i \]

\[ \mathbf{W}^o \]

\[ \mathbf{W}^h \]

**Jordan Network**

\[ y_t \]

\[ x_t \]

\[ \mathbf{W}^i \]

\[ \mathbf{W}^o \]

\[ \mathbf{W}^h \]
Deep RNN

\[
\begin{align*}
y^t & \\
& \uparrow \\
x^t & \\
& \rightarrow \rightarrow \\
y^{t+1} & \\
& \uparrow \\
x^{t+1} & \\
& \rightarrow \rightarrow \\
y^{t+2} & \\
& \uparrow \\
x^{t+2} & \\
& \rightarrow \rightarrow \\
\end{align*}
\]
Bidirectional RNN

\[ \begin{align*}
    x^t & \rightarrow y^t & \rightarrow x^{t+1} \\
    x^{t+1} & \rightarrow y^{t+1} & \rightarrow x^{t+2} \\
    \vdots & \vdots & \vdots
\end{align*} \]
Many to one

• Input is a vector sequence, but output is only one vector

**Sentiment Analysis**

- Positive (正雷)
- Negative (負雷)
- Positive (正雷)

我覺得很糟糕......
這部電影太糟糕了......
這部電影很棒......
Many to Many (Output is shorter)

- Both input and output are vector sequences, **but the output is shorter.**

*Speech Recognition*
Many to Many (Output is shorter)

• Both input and output are vector sequences, *but the output is shorter.*
• Connectionist Temporal Classification (CTC)
  • Add an extra symbol “φ” (同上)

好 φ φ 棒 φ φ φ φ φ  →  “好棒”

好 φ φ 棒 φ 棒 φ φ  →  “好棒棒”
Many to Many (No Limitation)

• Both input and output are vector sequences with different lengths. → Sequence to sequence learning

Machine Translation
Many to Many (No Limitation)

• 推文接龍
  • Ref: http://pttpedia.pixnet.net/blog/post/168133002-%E6%8E%A5%E9%BE%8D%E6%8E%A8%E6%96%87
Many to Many (No Limitation)

• Both input and output are vector sequences with different lengths. → Sequence to sequence learning

Add a symbol “===“ (斷)
One to Many

• Input is one vector, but output is a vector sequence

Caption generation
Outline

Vanilla Recurrent Neural Network (RNN)

Variants of RNN

Long Short-term Memory (LSTM)
Long Short-term Memory (LSTM)

- Input Gate
- Output Gate
- Forget Gate
- Memory Cell

Signal control
the input gate
(Other part of
the network)

Signal control
the output gate
(Other part of
the network)

Signal control
the forget gate
(Other part of
the network)

Special Neuron: 4 inputs, 1 output

Other part of the network
Activation function $f$ is usually a sigmoid function.

Between 0 and 1

Mimic open and close gate

$$c' = g(z)f(z_i) + cf(z_f)$$
Original Network:

- Simply replace the neurons with LSTM
4 times of parameters

\[ x_1 \quad x_2 \quad \text{Input} \]
When $x_2 = 1$, add the numbers of $x_1$ into the memory

When $x_2 = -1$, reset the memory

When $x_3 = 1$, output the number in the memory.
What is the next wave?

• Attention-based Model
Recommended Reading List

• The Unreasonable Effectiveness of Recurrent Neural Networks
  • http://karpathy.github.io/2015/05/21/rnn-effectiveness/

• Understanding LSTM Networks
  • http://colah.github.io/posts/2015-08-Understanding-LSTMs/
Acknowledgement

• 感謝 葉軒銘 同學於上課時發現投影片上的錯誤