Deep Learning for Business

Deep Learning with CNN & RNN

Deep Learning with RNN

(Recurrent Neural Network)

RNN (Recurrent Neural Network)

RNN Applications

- Speech Recognition
  - Apple's Siri
  - Google's Voice Search
  - Samsung's S Voice
- Handwriting Recognition
- Sequence Data Analysis
- Program Code Generation
  - RNN automatically generates computer programming code that can serve a predefined functional objective

RNN (Recurrent Neural Network)

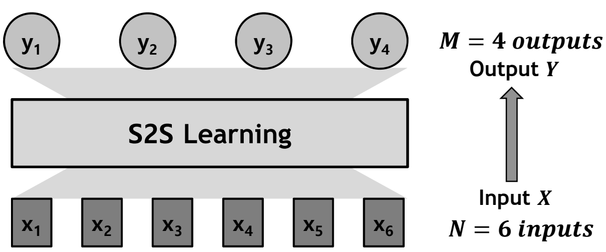
RNN Structure

- NN that uses directed cyclic connections between neurons
- Directed cyclic connections create internal states with dynamic temporal characteristics
- Internal memory is used to process arbitrary input data sequences
- Sequence Modeling is used for data sequence Classification & Clustering

RNN (Recurrent Neural Network)

RNN Structure

- Sequence Modeling structure based on S2S (Sequence to Sequence) Learning
  - *N* inputs are transformed into *M* outputs



RNN (Recurrent Neural Network)

RNN Process

1. Data input to the Input Layer

2. Representation of the data in the Input Layer is computed and sent to the Hidden Layer

3. Hidden Layer conducts Sequence Modeling and Training in Forward or Backward directions

RNN (Recurrent Neural Network)

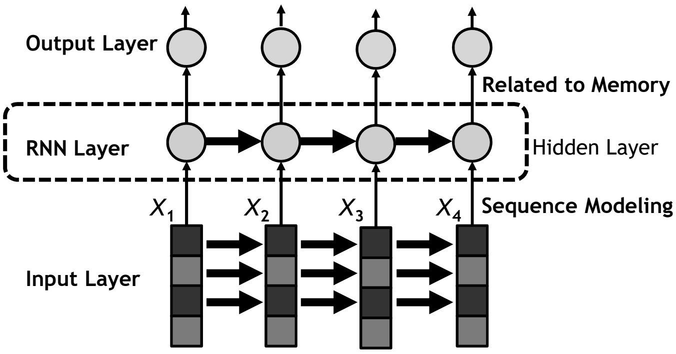
RNN Process

4. Multiple Hidden Layers using Forward or Backward direction Sequence Modeling and Training can be used

5. Final Hidden Layer sends the processed result to the Output Layer

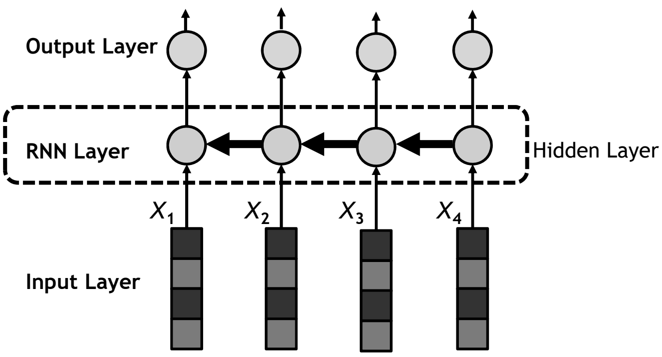
RNN (Recurrent Neural Network)

Forward RNN



RNN (Recurrent Neural Network)

Backward RNN



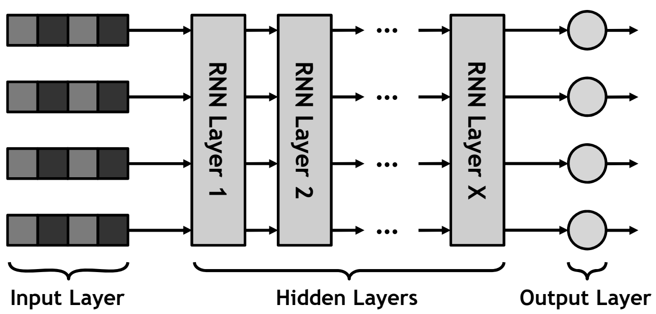
RNN (Recurrent Neural Network)

S2S (Sequence to Sequence)
Deep Learning RNN

– Assuming an input data sequence of
  Data 1, Data 2, Data 3, …
  applied to the RNN, a Hidden Layer may be in the form of a Forward RNN or Backward RNN

RNN (Recurrent Neural Network)

S2S Deep Learning RNN



RNN (Recurrent Neural Network)

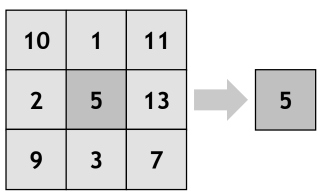
RNN Processes

– **Representation**
  - Applied to the Input Layer data
  - Similar to Subsampling (Pooling) in CNN
  - Representation is used to extract the important data that characterizes the data set in the best way
  - Representation is a non-linear down-sampling process

RNN (Recurrent Neural Network)

Representation Techniques

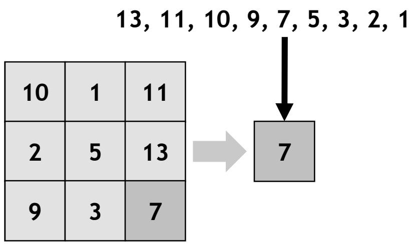
– **Center**
  - Center value is selected



RNN (Recurrent Neural Network)

Representation Techniques

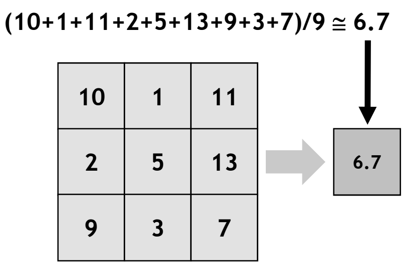
– **Median**
  - After lining up the data sequence from Largest to Smallest, the middle value is selected



RNN (Recurrent Neural Network)

Representation Techniques

– **Average**
  - Average value is used

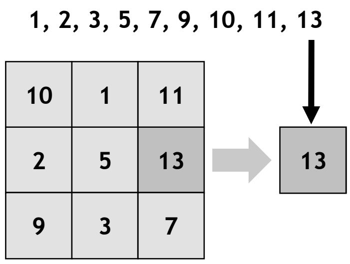


RNN (Recurrent Neural Network)

Representation Techniques

– **Max Pooling**

- Max value is selected

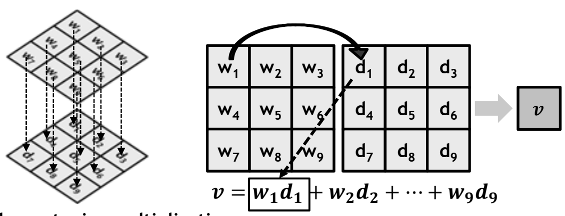


RNN (Recurrent Neural Network)

Representation Techniques

– **Weighted Sum**

- **Data values d_1, d_2,…, d_9 are scaled by weights w_1, w_2,…, w_9 to create the weighted sum $v = (w_1d_1 + w_2d_2 + \ldots + w_9d_9)$**



Element-wise multiplication

RNN (Recurrent Neural Network)

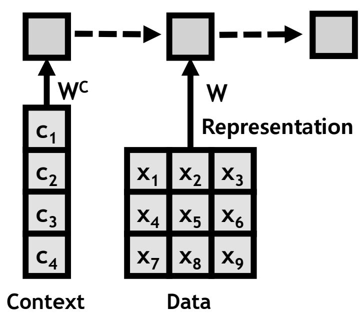
Context based Projection

- Use assisting data (context) in the representation process

- Context data may be the Original Input data or some type of Biased or Transformed data

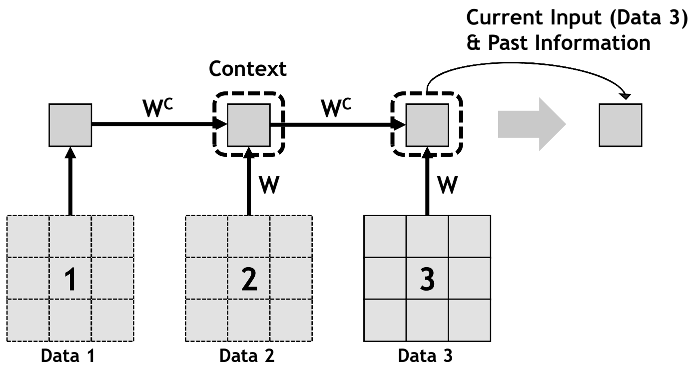
RNN (Recurrent Neural Network)

Context based Projection



RNN (Recurrent Neural Network)

Context based Projection



RNN (Recurrent Neural Network)

Representation with Attention

— Attention enables the decoder to attend to different parts of the source data segment at various steps of the output generation

— Language Translation Example
- RNN attends to sequential input states, multiple words simultaneously, and words in different orders when producing the output translated language

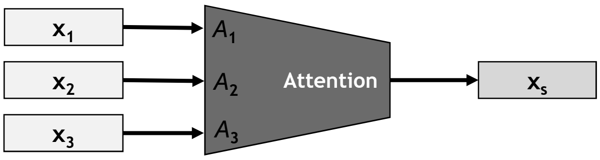
RNN (Recurrent Neural Network)

Representation with Attention

– SoftMax transfer
($\tilde{a}_1$ + $\tilde{a}_2$ + $\tilde{a}_3$) = 1

$$\begin{cases} a_1 \rightarrow \tilde{a}_1 \\ a_2 \rightarrow \tilde{a}_2 \\ a_3 \rightarrow \tilde{a}_3 \end{cases}$$

– Attention values (A_1, A_2, A_3) represent the importance of the data set



RNN (Recurrent Neural Network)

Representation with Attention

– SoftMax values (**$\tilde{a}_1$, $\tilde{a}_2$, $\tilde{a}_3$**) and Weights (**w_1, w_2, w_3**) and Attention values (**A_1, A_2, A_3**) are used to transform the values

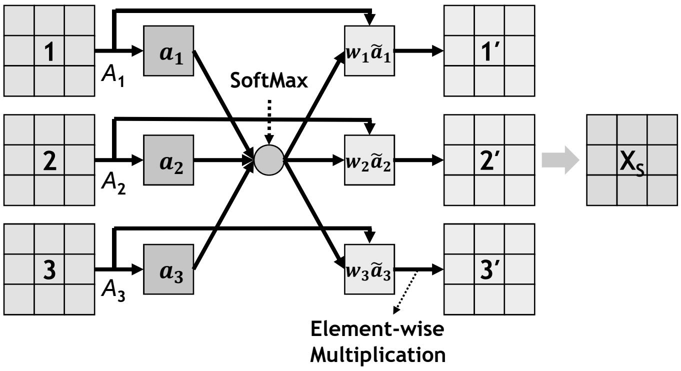
- $\tilde{a}_1 = \frac{e^{a_1}}{e^{a_1}+e^{a_2}+e^{a_3}}$
- $\tilde{a}_2 = \frac{e^{a_2}}{e^{a_1}+e^{a_2}+e^{a_3}}$
- $\tilde{a}_3 = \frac{e^{a_3}}{e^{a_1}+e^{a_2}+e^{a_3}}$

($\tilde{a}_1$ + $\tilde{a}_2$ + $\tilde{a}_3$) = 1

– $w_1\tilde{a}_1$ and $w_2\tilde{a}_2$ and $w_3\tilde{a}_3$ are used to transform each data set

RNN (Recurrent Neural Network)

Representation with Attention Example



RNN (Recurrent Neural Network)

FRNN (Fully Recurrent NN)

– RNN early model

– All neurons have connections to other neurons with modifiable weights

– Neurons form Input, Hidden, and Output Layers

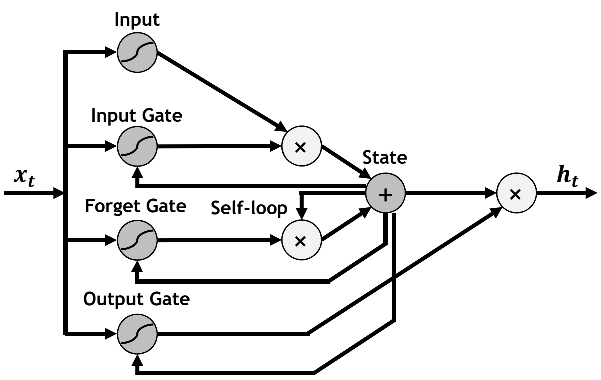
RNN (Recurrent Neural Network)

LSTM (Long Short-Term Memory) RNN

- Currently popular RNN model
- LSTM Cell has an Input Gate, Output Gate, Forget Gate, and Self-Loop (may have more gates)
- Self-Loop in the LSTM Cell gives a data sequence memory effect

RNN (Recurrent Neural Network)

LSTM RNN Cell Structure Example



RNN (Recurrent Neural Network)

LSTM (Long Short-Term Memory) RNN

– RNN Recurrent Gate (Forget Gate) effect
  - Prevents backpropagated errors from Vanishing (Vanishing Gradient Problem) or Exploding (Divergence Problem)
  - LSTM Recurrent Gates (Forget Gates) enable Errors to flow backwards through an unlimited number of VL (Virtual Layers) extending the memory characteristics

RNN (Recurrent Neural Network)

LSTM RNN Characteristics

– LSTM is effective on data sequences that require memory of far past events

  (e.g., thousands of discrete time steps ago)

– LSTM RNNs perform well on data sequences with long delays, and mixed signals with high and low frequency components

RNN (Recurrent Neural Network)

LSTM RNN Applications

- Speech Recognition and
  Large-Vocabulary Speech Recognition
- Pattern Recognition
- Connected Handwriting Recognition
- Text-to-Speech Synthesis

RNN (Recurrent Neural Network)

LSTM RNN Applications

- Recognition of Context Sensitive Languages
- Machine Translation
- Language Modeling
- Multilingual Language Processing
- Automatic image captioning
  (using LSTM RNN + CNN)

Deep Learning for Business

Deep Learning with CNN & RNN

References

References

- I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. book in preparation, MIT Press, www.deeplearningbook.org, 2016.
- D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel & S. Dieleman, “Mastering the game of Go with deep neural networks and tree search,” Nature, vol. 529, no. 7587, pp. 484-489, 28 Jan. 2016.
- Jared Hillam, "What is Hadoop?: SQL Comparison," YouTube, https://www.youtube.com/watch?v=MfF750YVDxM
- Wikipedia, http://www.wikipedia.org
- Nvidia, https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

References

Image sources

- Neuron
  By Dhp1080, svg adaptation by Actam (Image:Neuron.svg) [CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0) or GFDL (http://www.gnu.org/copyleft/fdl.html)], via Wikimedia Commons
- By Nicolas.Rougier (Own work) [GFDL (http://www.gnu.org/copyleft/fdl.html), CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/) or CC BY-SA 2.5-2.0-1.0 (http://creativecommons.org/licenses/by-sa/2.5-2.0-1.0)], via Wikimedia Commons
- By Copyright (c) 2003 Nicolas P. Rougier (Nicolas P. Rougier) [GPL (http://www.gnu.org/licenses/gpl.html)], via Wikimedia Commons

References

Image sources

- By Donarreiskoffer (Self-photographed) [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/)], via Wikimedia Commonsps://commons.wikimedia.org/w/index.php?curid=43383
- By David Lapetina (Own work) [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0)], via Wikimedia Commons