Intro to Recurrent Neural Networks

with Applications to NLP

Abay Bektursun
http://abay.tech/

Outline

- What are RNNs? - 2 min
- Why are they useful? - 8 min
- How do they work? - 30 min
- Revolution in NLP - 10 min

Music Generation

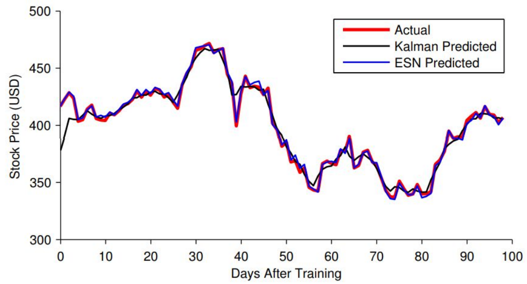
Performance RNN: Generating Music with Expressive Timing and Dynamics

DeadSeanKennedy

The 1st AI-composed concert (Sony Flow Machines)

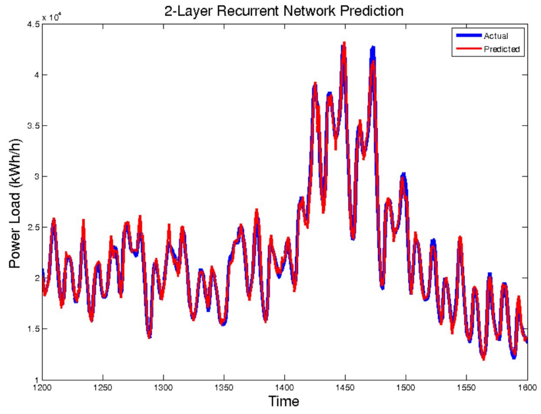
Forecasting

ESN was tested on Google's stock price in comparison to the Kalman filter (see Figure 5). The ESN captures quick changes in the stock price whereas the simple Kalman filter cannot.



Source: Armando B. et al., Financial Market Time Series Prediction with Recurrent Neural Networks

Figure 1: Load prediction with recurrent neural network



Source: Enzo Busseti, et al., Deep Learning for Time Series Modeling

Image Captioning

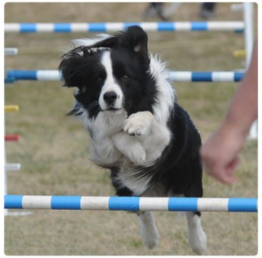
"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

"girl in pink dress is jumping in air."



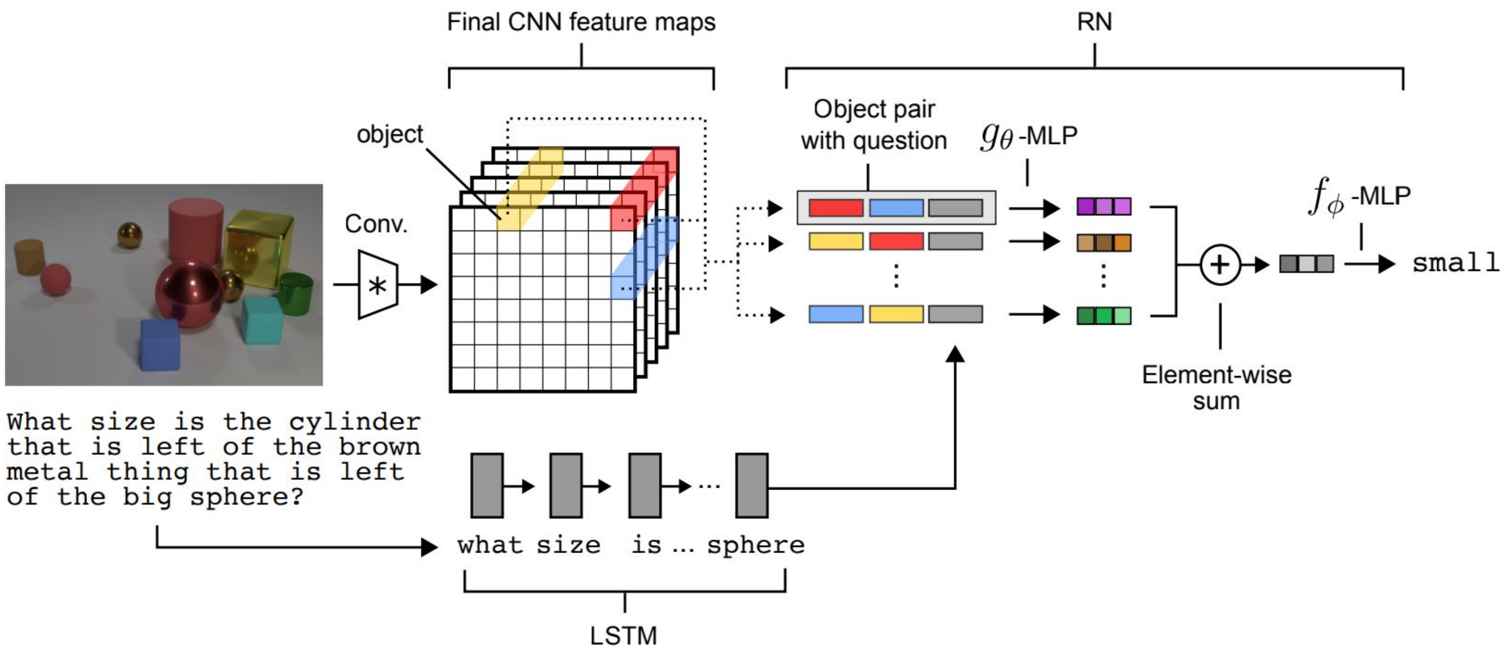
"black and white dog jumps over bar."

"young girl in pink shirt is swinging on swing."

"man in blue wetsuit is surfing on wave."

Source: http://cs.stanford.edu/people/karpathy/deepimagesent/

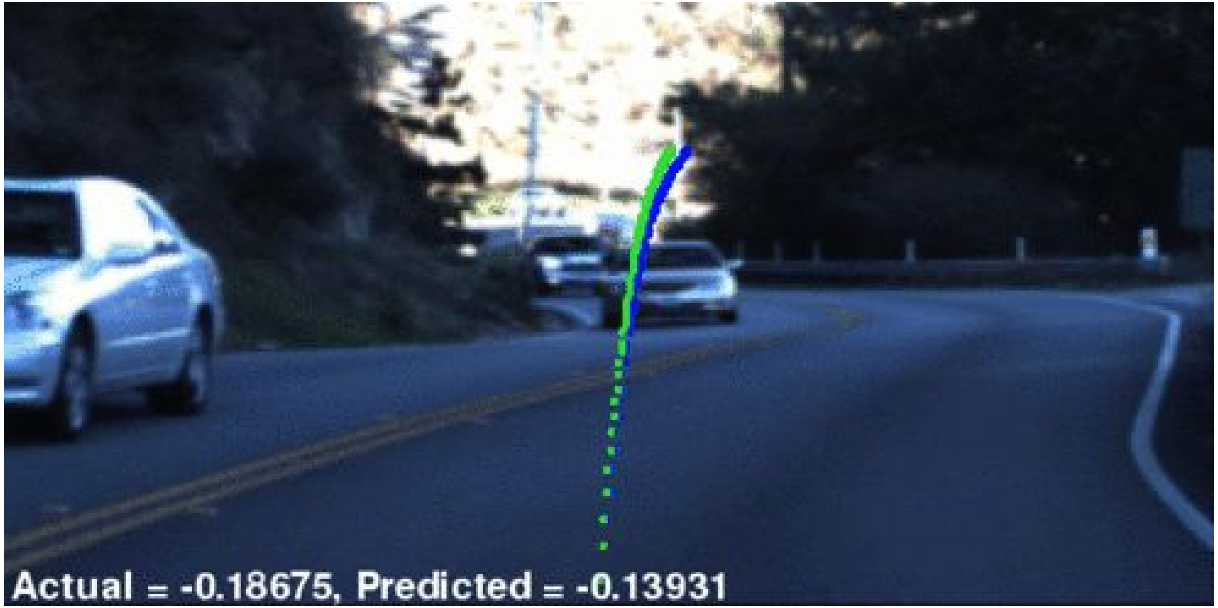
Relational Reasoning



Source: https://deepmind.com/blog/neural-approach-relational-reasoning/

Self Driving Cars

Udacity Challenge 1st Place Winner: Team Komanda



Source: https://github.com/udacity/self-driving-car/tree/master/steering-models/community-models

What can you put in and get out?

Sequence to Vector

Vector to Sequence

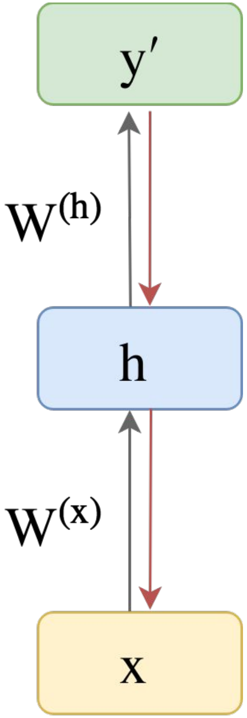
Sequence to Sequence

Sequence to Sequence (Synced)

Nothing to Sequence

Let's Formulate RNNs

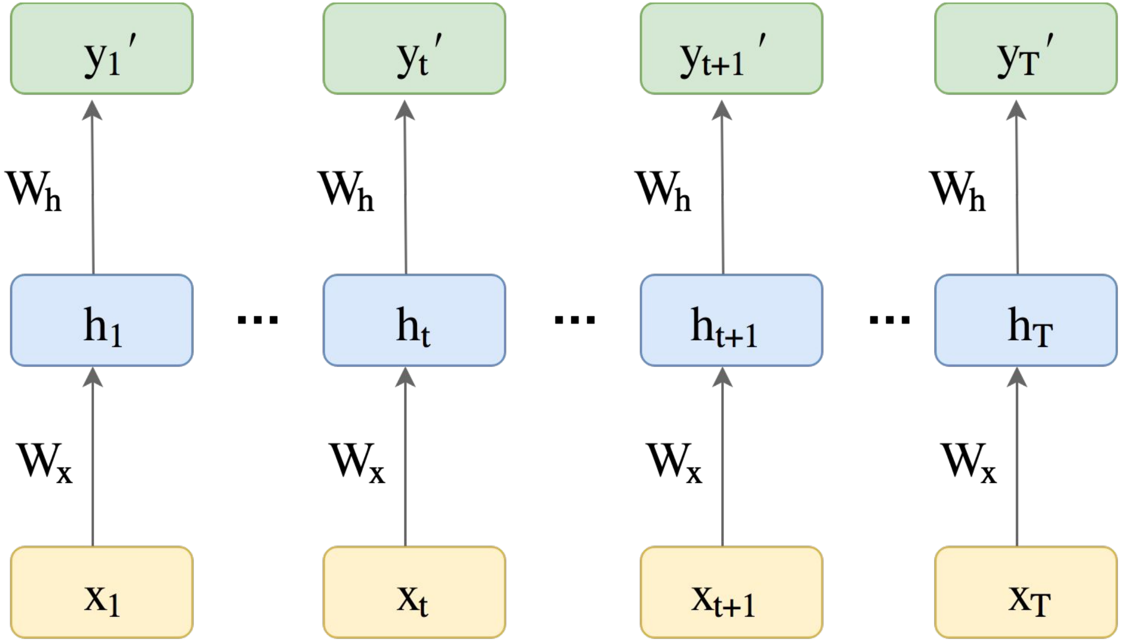
Forward Pass in NNs



$$y' = \sigma(W^{(h)}h + c)$$

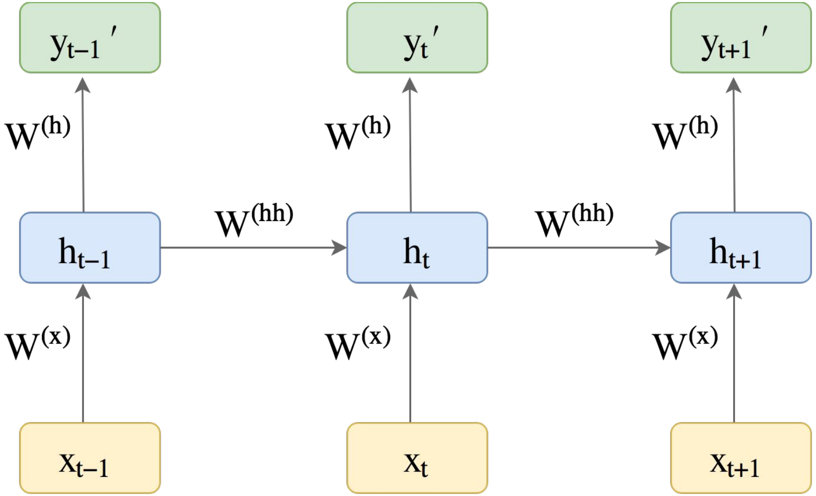
$$h = \sigma(W^{(x)}x + b)$$

$$x = data$$



y_1'
y_t'
y_{t+1}'
y_T'
W_h
W_h
W_h
W_h
h_1
...
h_t
...
h_{t+1}
...
h_T
W_x
W_x
W_x
W_x
x_1
x_t
x_{t+1}
x_T

Basic RNN



$$y_t' = \sigma(W^{(h)} h_t + c)$$

$$h_t = \sigma(W^{(hh)} h_{t-1} + W^{(x)} x_t + b)$$

$$x_1, \ldots, x_{t-1}, x_t, \ldots, x_T$$

Sequence of vectors (Data)

Basic RNN

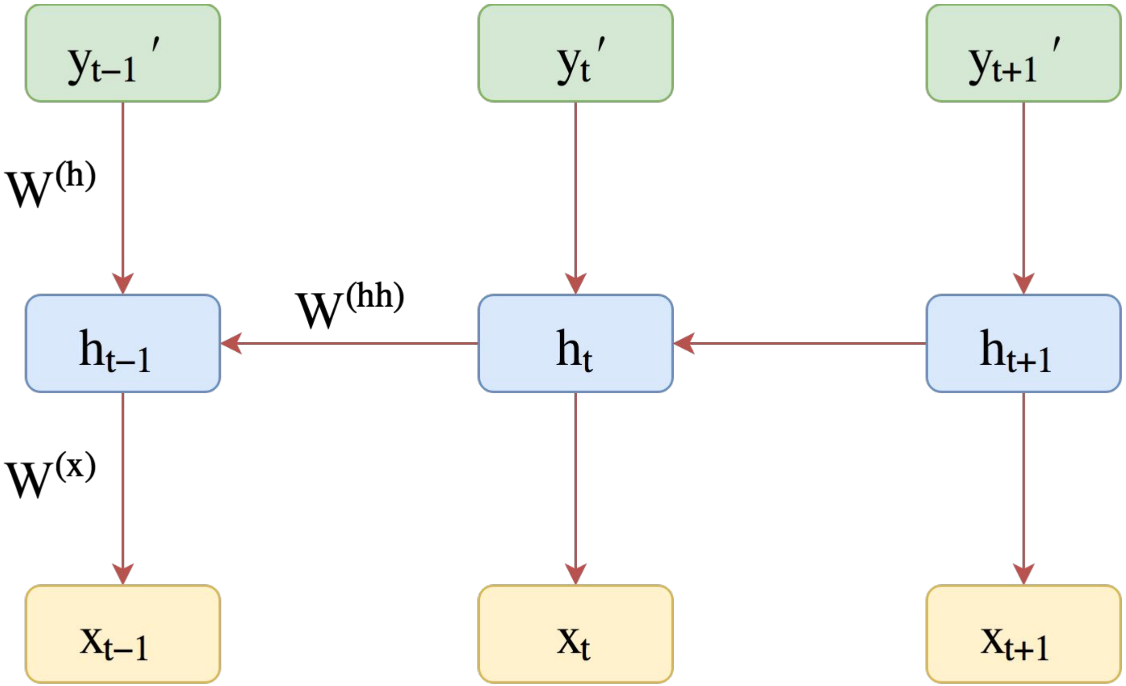
$$y'_t = softmax(W^{(h)} h_t + c)$$

$$h_t = \sigma(W^{(hh)} h_{t-1} + W^{(x)} x_t + b)$$

Sequence of vectors (Data):

$$x_1, \ldots, x_{t-1}, x_t, \ldots, x_T$$

RNN Backpropagation



RNN Backpropagation

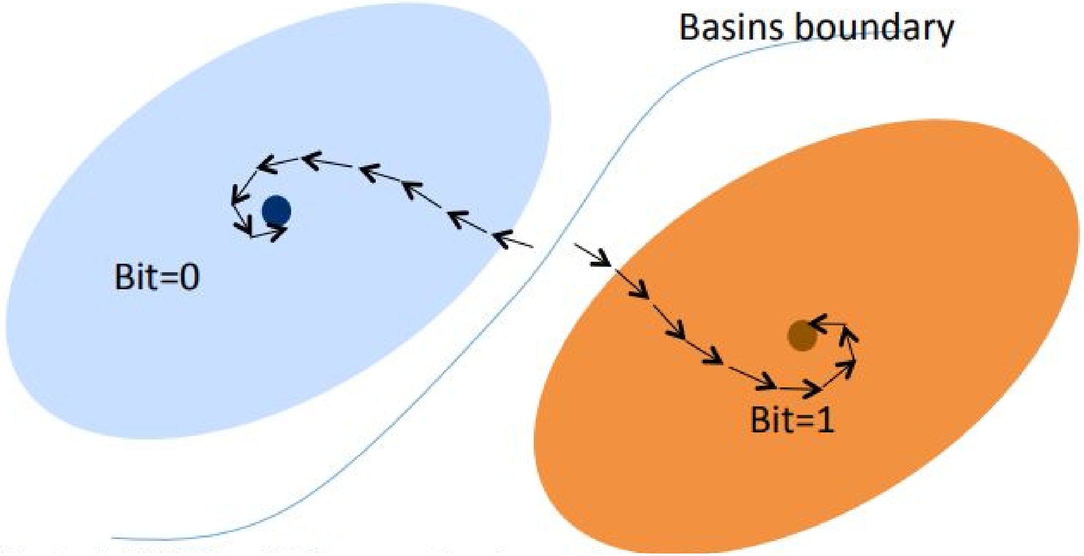
Loss: $L(\{x_1, .., x_T\}, \{y_1, .., y_T\})$

Gradient: $$\frac{\partial L_t}{\partial W^{(hh)}} = \sum_{k=1}^{t} \frac{\partial L_t}{\partial y'_t} \frac{\partial y'_t}{\partial h_t} \boxed{\frac{\partial h_t}{\partial h_k}} \frac{\partial h_k}{\partial W^{(hh)}}$$

$$\boxed{\frac{\partial h_t}{\partial h_k}} = \prod_{i=k+1}^{t} \frac{\partial h_i}{\partial h_{i-1}}$$

$W^{(hh)} \in \mathbb{R}^{D_h \times D_h}$ $W^{(hx)} \in \mathbb{R}^{D_h \times d}$ $W^{(S)} \in \mathbb{R}^{|V| \times D_h}$

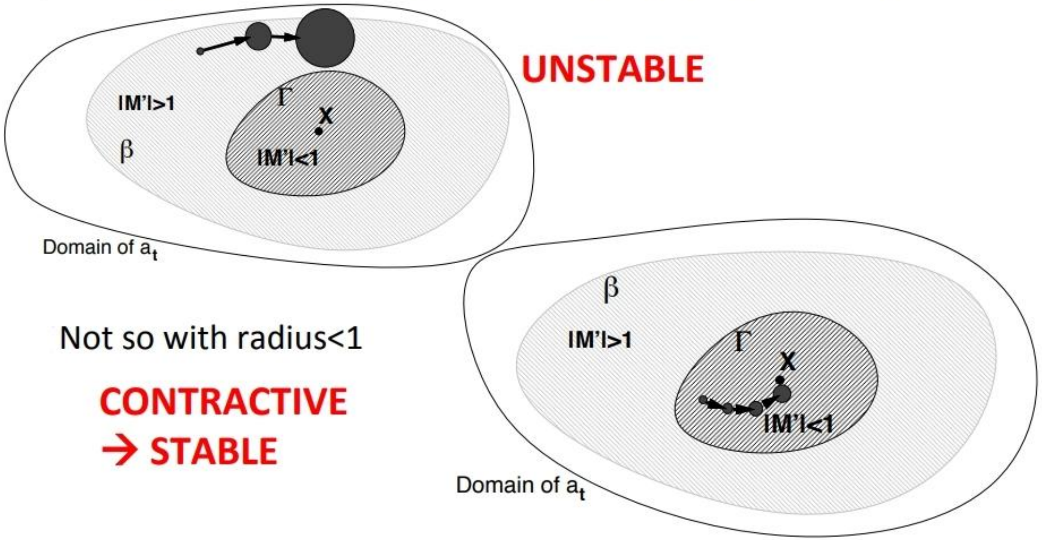
RNN is a Dynamical System



Source: https://www.youtube.com/watch?v=AYku9C9XoB8

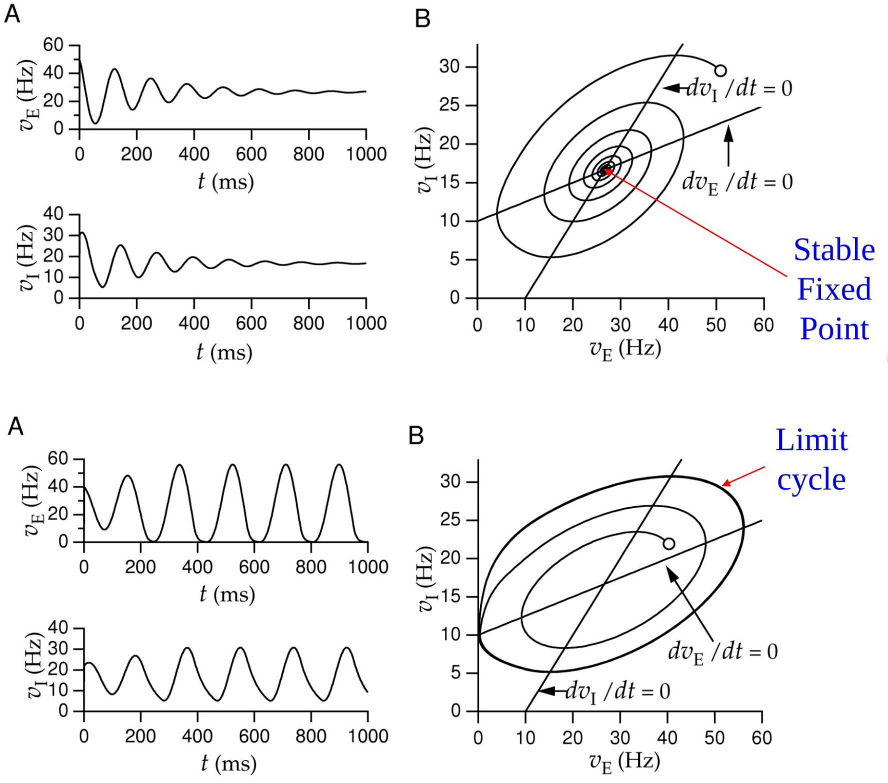
RNN is a Dynamical System

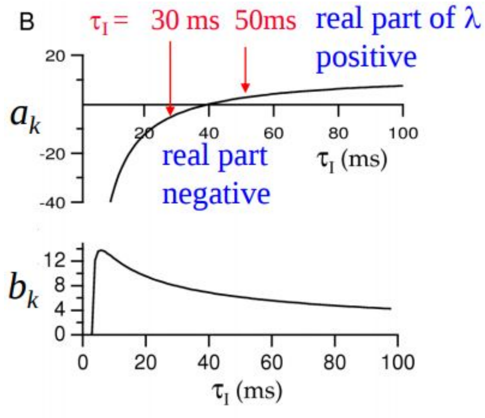
With spectral radius > 1, noise can kick state out of attractor



Source: https://www.youtube.com/watch?v=AYku9C9XoB8

Neuroscientific View of Recurrent Dynamics





Jacobian Matrix:

$$J = \begin{bmatrix} \dfrac{(M_{EE}-1)}{\tau_E} & \dfrac{M_{EI}}{\tau_E} \\ \dfrac{M_{IE}}{\tau_I} & \dfrac{(M_{II}-1)}{\tau_I} \end{bmatrix}$$

Gated Units (GRU)

Reset Gate: $\boxed{r_t} = \sigma(W^{(r)}x_t + U^{(r)}h_{t-1})$

Update Gate: $\boxed{z_t} = \sigma(W^{(z)}x_t + U^{(z)}h_{t-1})$

Cell: $h_t = \boxed{z_t} \odot h_{t-1} + (1 - \boxed{z_t}) \odot tanh(Wx_t + \boxed{r_t} \odot Uh_{t-1})$

Practical Advices

- Initialize parameters with identity matrices
- Use gradient clipping
- Use gated cells
- Use ReLU to combat instability
- Always save your parameters during training

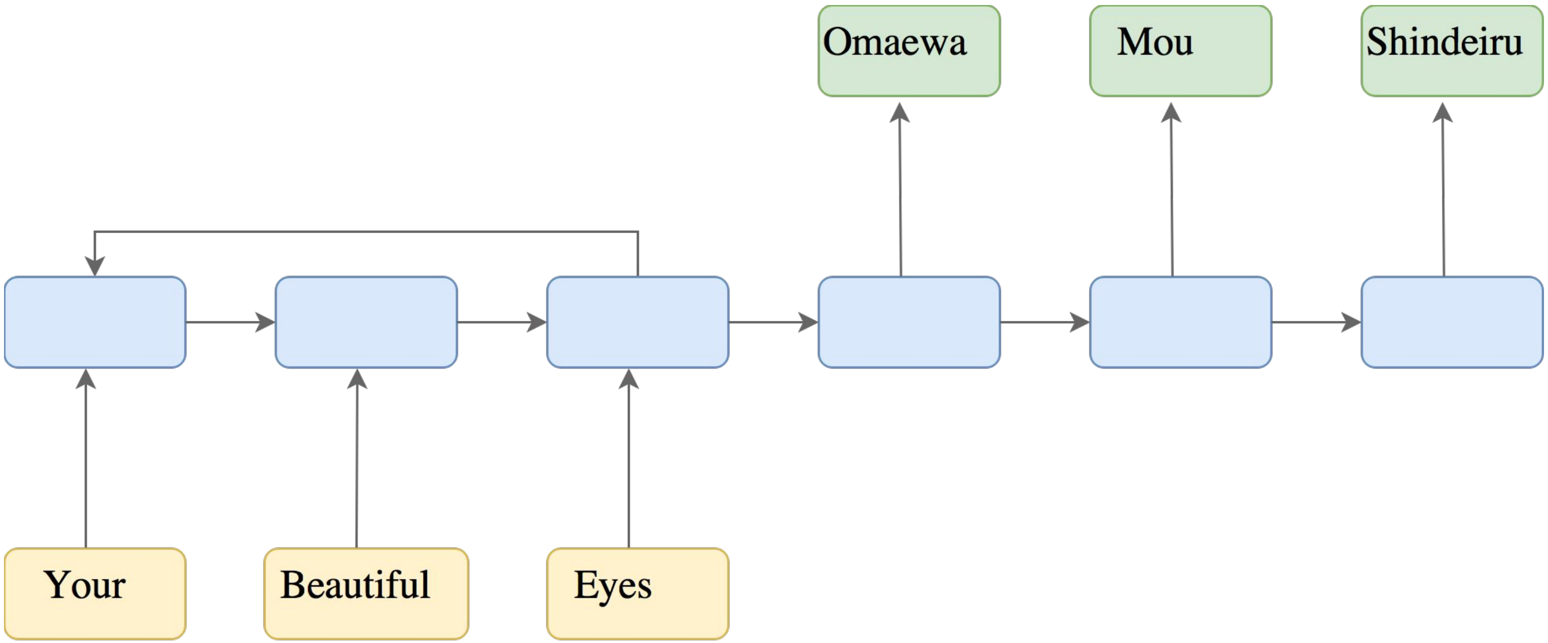
Natural Language Processing

Traditional NLP:

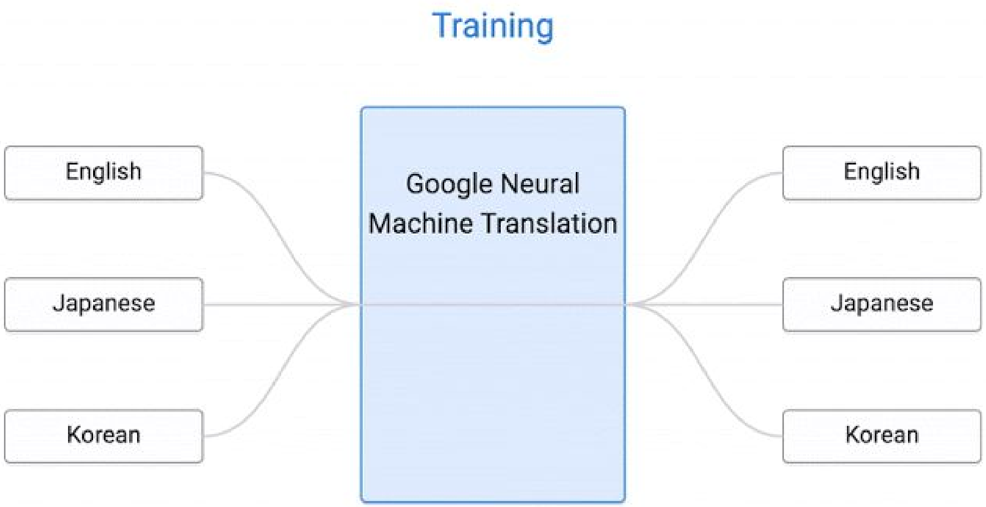
- Incorrect Markov assumptions
- Inflexible
- Short memory
- Probabilities computed with unigrams and bigrams. Computationally unrealistic for practical purposes

Sources: https://deepmind.com/blog/neural-approach-relational-reasoning/; Wu et al, Nature, 2016

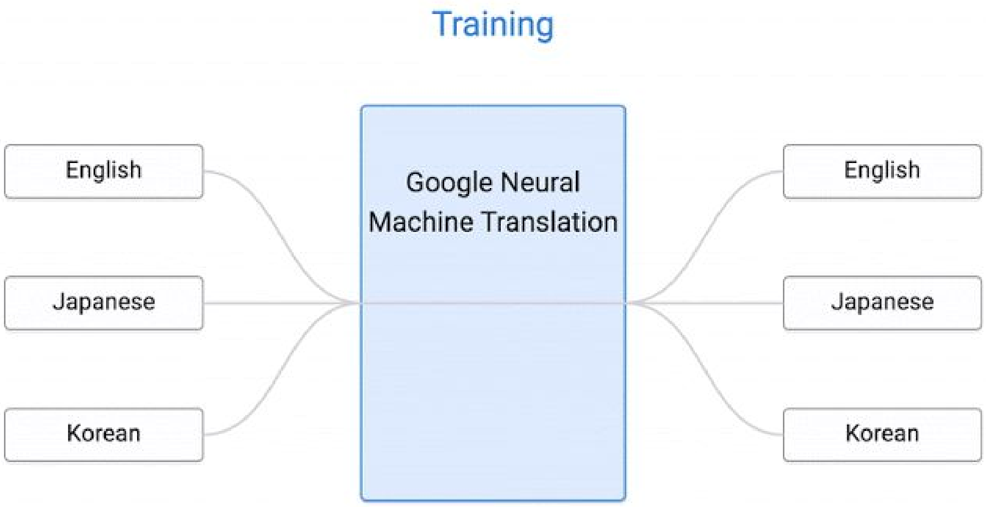
Translation



Google Translator



Google Translator



<2es> Hello, how are you? -> ¿Hola como estás?

My Top Picks of Researchers

Yoshua Bengio

- http://www.iro.umontreal.ca/~bengioy/yoshua_en/

Richard Socher

- http://www.socher.org/

Andrej Karpathy

- http://cs.stanford.edu/people/karpathy/

References

- DeepMind (Jun 2017), A neural approach to relational reasoning (https://deepmind.com/blog/neural-approach-relational-reasoning/)
- Christian W. Omlin ,C. Lee Giles, Symbolic Knowledge Representation in Recurrent Neural Networks: Insights from Theoretical Models of Computation
- CS224n: Natural Language Processing with Deep Learning. Retrieved November 20, 2017, from http://web.stanford.edu/class/cs224n/
- Goodfellow-et-al-2016,Deep Learning,MIT Press, 2016: http://www.deeplearningbook.org
- Prof. Yoshua Bengio - Recurrent Neural Networks (RNNs), https://www.youtube.com/watch?v=AYku9C9XoB8
- Andrej Karpathy, The Unreasonable Effectiveness of Recurrent Neural Networks: http://karpathy.github.io/2015/05/21/rnn-effectiveness/
- Christopher Olah, Understanding LSTM Networks: http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- Enzo Busseti, Ian Osband, Scott Wong, Deep Learning for Time Series Modeling, December 14th, 2012
- Armando Bernal, Sam Fok, Rohit Pidaparthi, Financial Market Time Series Prediction with Recurrent Neural Networks
- Andrej Karpathy, Li Fei-Fei, Deep Visual-Semantic Alignments for Generating Image Descriptions
- Computational Neuroscience, https://www.coursera.org/learn/computational-neuroscience
- Wu et al, Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, Nature, 2016