CS577 : AS5

Theo Guidroz

1) [The quick brown fox jumped over the lazy dog]

	the	quick	brown	fox	jumped	over	the	lazy	dog
the	1	0	0	0	0	0	1	0	0
quick	0	1	0	0	0	0	0	0	0
brown	0	0	1	0	0	0	0	0	0
fox	0	0	0	1	0	0	0	0	0
jumped	0	0	0	0	1	0	0	0	0
over	0	0	0	0	0	1	0	0	0
lazy	0	0	0	0	0	0	0	1	0
dog	0	0	0	0	0	0	0	0	1

2)

	animal	color	action
the	0.2	0.1	0.1
quick	0.01	0.05	0.4
brown	0.3	0.95	0.6
fox	0.9	0.4	0.1
jumped	0.05	0.2	0.9
over	0.2	0.1	0.3
lazy	0.3	0.2	0.4
dog	0.95	0.3	0.4

3) $i_t = \sigma [(state_t \cdot Ui) + (input_t \cdot Wi) + bi]$

$f_t = \sigma [(state_t \cdot Uf) + (input_t \cdot Wf) + bf]$] gate

$O_t = \sigma [(state_t \cdot UO) + (input_t \cdot WO) + bo]$

$c_t = i_t \times b_t \times c_t \times f_t$] carry

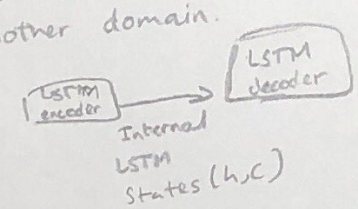
$state_t = O_t \times \tanh(c_t)$] state

4) The output of input, forget and output gates must be in the range 0 to 1 while the block gate in the range -1 to 1.

5) The problems with simple RNN is that it has problems retaining information and vanishing gradients causes long term dependencies. LSTM adds a way to carry information accross many timesteps.

6) Bidirectional will learn a different representation during reverse training and this can produce a stronger model. Bidirectional RNN performs better on sentiment analysis but worse on temperature prediction

7) It is uses to convert sequences from one domain to sequences in another domain.



The training set includes the source sequence which is fed to the encoder, the target sequence to the decoder input and the target sequence shifted forward to the decoder output. Cross-entropy is used as loss function. The encoder's hidden states summarize the input and produces a probability distribution over entire dictionary as output.