1. The quick brown fox jumped over the lazy dog.

Rearranging the terms in alphabetical order and created a one hot encoding

1	2	3	4	5	6	7	8	
[0	0	0	0	0	0	0	1]	the
[0	0	0	0	0	0	1	0]	quick
[1	0	0	0	0	0	0	0]	brown
[0	0	1	0	0	0	0	0]	fox
[0	0	0	1	0	0	0	0]	jumped
[0	0	0	0	0	1	0	0]	over
[0	0	0	0	0	0	0	1]	the
[0	0	0	0	1	0	0	0]	lazy
[0	1	0	0	0	0	0	0]	dog

2.

	Animal	color	action
The	0.02	0.01	0.01
quick	0.04	0.03	0.88
brown	0.07	0.92	0.01
fox	0.92	0.01	0.02
jumped	0.08	0.06	0.89
over	0.04	0.06	0.05
lazy	0.05	0.07	0.08
dog	0.91	0.01	0.04

3.

- Input gate: $\sigma_1(w_i[h_{t-1}, x_t] + b_i)$
  Forget gate: $\sigma_1(w_f[h_{t-1}, x_t] + b_f)$
  Output gate: $\sigma_1(w_o[h_{t-1}, x_t] + b_o)$
  Carry gate: $\sigma_2(w_c[h_{t-1}, x_t] + b_c)$
  Carry state = i/p gate * carry + carry state * forget gate
  State_t = o/p gate * σ_2(carry state)

where σ_1 = Sigmoid σ_2 = tanh.

- Sigmoid function is used in input, output and forget gates. It returns two values [0, 1]. Thus when $f_t = 0$ → forgets, $f_t = 1$ → remembers and so sigmoid fn is used.

~~For the carry fn tanh fn is used which returns values [-1, 1].~~

Block gate uses tanh fn that returns values from [-1, 1].

5. Problems of RNN
   1) Does not learn long term dependencies well
   2) Gradients get vanished
   3) Problem with retaining info seen many times.

LSTM:

Adds carry gate to remember info across timestamps.

Thus solves vanishing gradients issue

6. Bidirectional RNN:

Processes both in chronologi and in reverse chronological ways.

Thus it can be used in applications where sequences is important

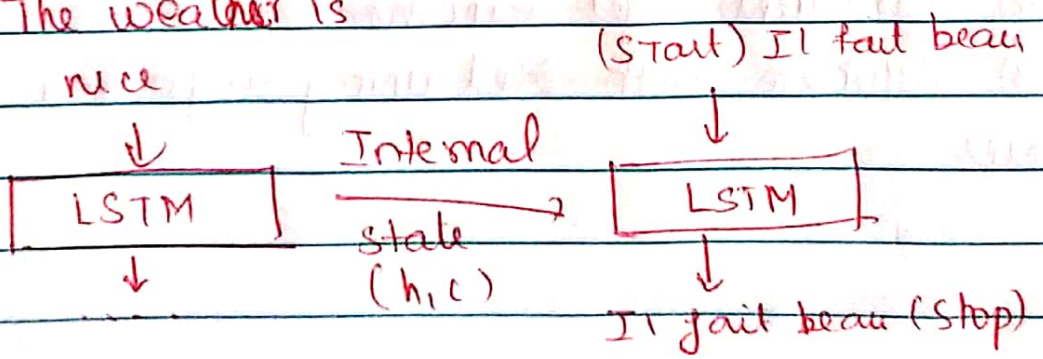
Use: Natural Lang. Processing
Text Analytics

Worse: Temperature Prediction

7. Seq2Seq:

Converts sequences from one lang to other domain

Encoder:



Training

Encoder is trained with the inputs which consists of sequences of 1 domain

Decoder is trained with the input that has sequences of ~~domain~~ other domain & output with same sequences shifted 1 timestamp.

The loss fn is calculated by the difference b/w the o/p of 1 LSTM block of the decoder and the o/p of the 2nd LSTM block of the decoder.