Ans to the Qs: 3

To solve the vanishing gradient problem of a standard RNN, GRU uses the two gates - update gate and reset gate. Basically, these are two vectors which decide what information should be passed to the output

Update Gate: The update gate helps the model to determine how much of the past information from previous time steps needs to be passed along to the future.

~~Reset gate: The reset gate is used from the model to decide how much~~

For a fully ~~connected~~ gated unit,

$$z_t = \sigma_g (W_z \cdot x_t + U_z \cdot h_{t-1} + b_z)$$

Reset Gate: The reset gate is used from the model to decide how much of the past information to forget.

For a fully gated unit,

$$r_t = \sigma_g (W_r \cdot x_t + U_r \cdot h_{t-1} + b_r)$$

The key differences between GRU and LSTM is given below:

1) GRU has 2 gates - update and reset gates. LSTM has 3 gates - input, output, and forget gates.

2) GRU is less complex than LSTM as it ~~has~~ has less number of gates.

3) If the dataset is small then GRU is preferred. Otherwise, LSTM should be ~~use~~ used for larger dataset.

4) LSTM has cell memory whereas GRU doesn't. LSTM used cell memory to avoid vanishing gradient problem whereas GRU uses computational data flow.

5) GRU improved LSTM by omitting cell memory and using fewer parameters.

6) GRU exposes the complete memory and hidden layers but LSTM doesn't.

RNNs face short-term memory problem. It is caused due to vanishing gradient problem. As RNN processes more steps it suffers from vanishing gradient more than other neural network architectures. RNNs also face the counter-part of vanishing gradient problem called the exploding gradient problem. GRUs do not face such issues.

GRU is faster than RNN as it uses less parameters. GRU can be made even faster using minimal gated variant which uses only one gate which is a combination of update and reset gates of the fully gated variant.