GRU – Gated recurrent unit.

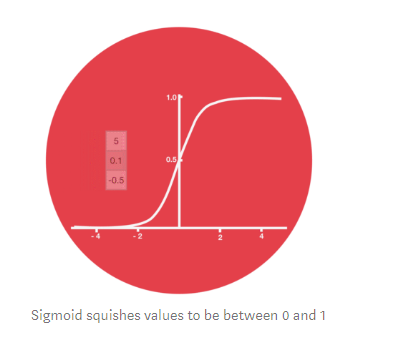
1: Forget or reset gate

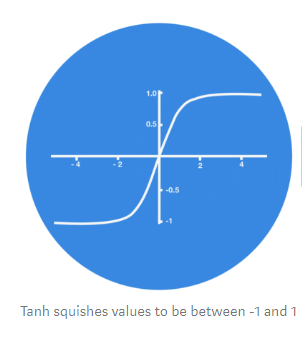
2: update gate.

Gates uses sigmoid functions with point wise multiplication.

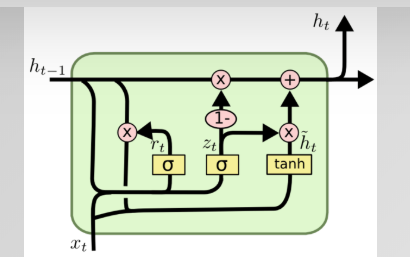
GRU has fewer parameters (U and W are smaller) hence it trains faster and needs less data to generalize.

Sigmoid functions are used so that the output can be compressed between 0 and 1.

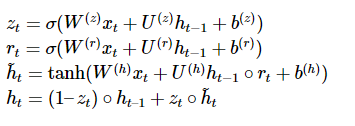




GRU’s are used to overcome the vanishing gradient problem faced by RNN (Recurrent Neural Network)

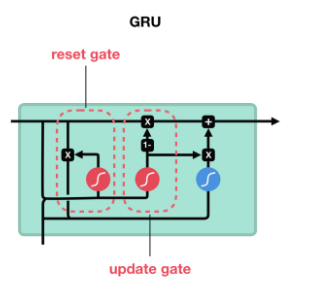


Mathematical GRU Model.



Zt filter for previous state, Zt lower then lot of previous state is reused, So the input Xt of current state does not affect the output , if Zt is high then output of the current step is influenced a lot by the current input step Xt but it is not influenced a lot by the previous state Ht-1

Rt is the forget gate or reset gate, it allows the cell to forget certain parts of the state.



Example:

from keras.models import Sequential

from keras.layers.recurrent import GRU

|  |
| --- |
| model = Sequential() |
|  | model.add(GRU(X\_train.shape[-1], y\_train.shape[-1])) |
|  | model.add(Activation('softmax')) |
|  | model.compile(loss='categorical\_crossentropy', optimizer='adadelta') |
|  | history = model.fit(X\_train, y\_train, nb\_epoch=12, batch\_size=16, validation\_data=(X\_test, y\_test), show\_accuracy=True, verbose=2) |

Reference Link:

1: <https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>

2: <https://www.data-blogger.com/2017/08/27/gru-implementation-tensorflow/>