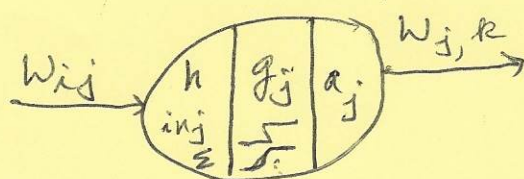


2 NOV '16

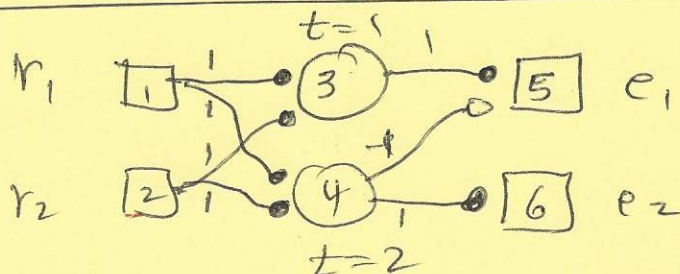
Recurrent NN Computation Model



$g_j(a_j)$: performed as uniquely assigned function at each node
 $W\bar{a}$: performed as matrix multiply of fully connected network

stopping criterion ? feedforward : # layers
 loops : ?

Eg.,



$$W = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$g_i = id \quad i=1,2,5,6$ don't encode these

$g_i = \text{thresh} \quad i=3,4$

$g(1).node = 3;$
 $g(2).node = 4;$

$g(1).g = 'TA - \overset{g^3}{\text{thresh}}';$
 $g(2).g = 'TA - g^4';$

2 NOV '16

function a_out = TA_RNN_step(W, a, g)

a_out
~~W~~ = W * a;

~~a_out~~

len_g = length(g);

for k = 1: len_g

node
~~index~~ = g(k).node;

g_k = g(k) * g;

a_out(node) = feval(g_k, a_out(node));

end

function a_out = TA_RNN_run(W, a, g, max_iter)

%

num_nodes =

a_out = a;

for c = 1: max_iter

a_out = TA_RNN_step(W, a_out, g);

end