① Neural turing Machine Neural Turing Machine

→ memory location.

Reading : memory $M_t \rightarrow \boxed{N} \times \boxed{M}$ matrix

↓ time

└→ vector size at each location.

Attention weight, $0 \le w_t(i) \le 1$; $i \sim \{1 \dots N\}$ with constraint $\boxed{\sum_i w_t(i) = 1}$

(1×M) size

→ i'th row

Read memory $r_t \leftarrow \boxed{\sum_i w_t(i)\, M_t(i)}$ → convex combination.

weighted sum of rows.

Differentiable.

Writing :

[1 1 ... 1] → erase vector (1×M)

$$\tilde{M}_t(i) \leftarrow \underbrace{M_{t-1}(i)\left[\mathbf{1} - w_t(i)\, e_t\right]}_{\text{pointwise multiply}}$$

After erase add

$$M_t(i) \leftarrow \tilde{M}_t(i) + w_t(i)\, a_t$$

→ both differentiable

2. Constructing weight vector:

Focusing by content

key strength.

cosine sim

key vector (1×M)

$$w_t^c(i) \leftarrow \frac{\exp\left(\beta_t K\left[k_t, M_t(i)\right]\right)}{\sum_j \exp\left(\beta_t K\left[k_t, M_t(j)\right]\right)}$$

focusing by location:

interpolation gate (0, 1)

$$\underline{w}_t^g \leftarrow g_t \underline{w}_t^c + (1 - g_t) \underline{w}_{t-1}$$

$$\tilde{w}_t(i) \leftarrow \sum_{j=0}^{N-1} w_t^g(j) \underbrace{s(i-j)}_{\text{shift weight}},$$

$$w_t(i) \leftarrow \frac{\tilde{w}_t(i)^{\gamma_t}}{\sum_i \cancel{\Sigma}\, w_t(i)^{\gamma_t}}$$

sharpening & Normalization.