On Generalization Bounds of a Family of Recurrent Neural Networks

Minshuo Chen *, Xingguo Li†, Tuo Zhao*
*Georgia Tech †Princeton University

Background

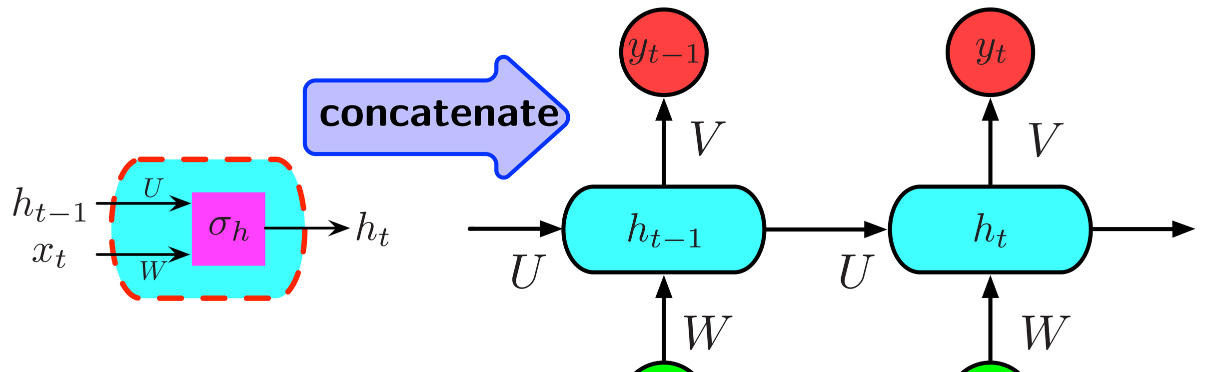
Vanilla RNNs iteratively compute $h_{i,t}$ and $y_{i,t}$ in a seq2seq classification problem,

$$h_{i,t} = \sigma_h\left(Uh_{i,t-1} + Wx_{i,t}\right), \quad \text{and} \quad y_{i,t} = \sigma_y\left(Vh_{i,t}\right),$$

- $(x_{i,t}, z_{i,t})_{t=1}^{T}$ is a sequence of data points. $z_{i,t} \in \{1, \ldots, K\}$ is the class label.
- σ_y and σ_h are activation operators.
- $h_{i,t}$ is the hidden state with $h_{i,0} = 0$. $y_{i,t}$ is the output signal.
- U, V, and W are weight matrices.

For a new testing sequence $(x_t, z_t)_{t=1}^{T}$, we predict the label sequence using

$$\widetilde{z}_t = \mathrm{argmax}_j [y_t]_j, \quad \text{for all } t = 1, \ldots, T.$$



Building Block

> **Questions**:
> - RNNs suffer from significant curse of dimensionality?
> - Advantages of MGU and LSTM over vanilla RNNs?

Problem Setup

Assumption 1 (Bounded Input). $\|x_{i,t}\|_2 \leq B_x$ *for all* i, t.

Assumption 2 (Bounded Spectral Norm). $\|U\|_2 \leq B_U$, $\|V\|_2 \leq B_V$, *and* $\|W\|_2 \leq B_W$.

Assumption 3 (Lipschitz Activation). σ_h *and* σ_y *are* ***1-Lipschitz*** *with* $\sigma_h(0) = \sigma_y(0) = 0$ *and* $\max_x \sigma_h(x) \leq b$.

Assumption 4 (Bounded $\ell_{2,1}$ Norm). $\|U\|_{2,1} \leq M_U$, $\|V\|_{2,1} \leq M_V$, *and* $\|W\|_{2,1} \leq M_W$.

Assumption 5 (Bounded Frobenius Norm). $\|U\|_F \leq B_{U,F}$, $\|V\|_F \leq B_{V,F}$, *and* $\|W\|_F \leq B_{W,F}$.

We denote

- **Function Class**: $\mathcal{F}_t = \{f_t : X_t \mapsto y_t\}$,
- **Margin**: $\mathcal{M}(f_t(X_t), z_t) = [f_t(X_t)]_{z_t} - \max_{j \neq z_t} [f_t(X_t)]_j$,
- **Ramp Risk**: $\widehat{\mathcal{R}}_\gamma(f_t) = \frac{1}{m}\sum_{i=1}^{m} \ell_\gamma\left(-\mathcal{M}(f_t(X_{i,t}), z_{i,t})\right)$, where ℓ_γ is the Ramp Loss with margin value γ.

Generalization Bound of Vanilla RNNs

We define *Model Complexity* of vanilla RNNs as

$$\textbf{Complexity} = d \times \Pi.$$

- d is the square root of **Number of Parameters**.
- $\Pi = B_V \min\left\{b\sqrt{d}, B_x B_W \sum_{i=0}^{t-1} B_U^i\right\}$ is the **Sum of Spectral Norm Products**.

Our generalization bound is stated in terms of complexity,

Theorem 1.

- *Activation operators σ_h and σ_y are given, and Assumptions 1–3 hold;*
- *$S = \left\{(x_{i,t}, z_{i,t})_{t=1}^{T}, i = 1, \ldots, m\right\}$ are drawn i.i.d. from any underlying data distribution.*

$\Longrightarrow$ *with probability at least $1 - \delta$ over S,*

$$\mathbb{P}\left(\widetilde{z}_t \neq z_t\right) - \widehat{\mathcal{R}}_\gamma(f_t) \leq \widetilde{O}\left(\frac{\textbf{Complexity}}{\sqrt{m}\gamma} + \sqrt{\frac{\log \frac{1}{\delta}}{m}}\right),$$

holds for any margin value $\gamma > 0$ and every $f_t \in \mathcal{F}_t$.

Differentiate the bound in 3 scenarios:

$$\left.\begin{array}{l} \bullet\ B_U < 1, \text{ the bound is } \widetilde{O}\left(\frac{d}{\sqrt{m}\gamma}\right) \\ \bullet\ B_U = 1, \text{ the bound is } \widetilde{O}\left(\frac{dt}{\sqrt{m}\gamma}\right) \\ \bullet\ B_U > 1, \text{ the bound is } \widetilde{O}\left(\frac{\sqrt{d^3 t}}{\sqrt{m}\gamma}\right) \end{array}\right\} \textbf{Polynomial} \text{ in } d, t.$$

> Complexity of Vanilla RNNs **does not suffer** from significant curse of dimensionality!

Compared to the generalization bound in [4],

$$\widetilde{O}\left(\frac{dt^2 B_W B_V \max\{1, B_U^t\}}{\sqrt{m}\gamma}\right),$$

our bound is **tighter** in all 3 scenarios.

Refined Generalization Bounds

Let $S_{2,1} = M_U + M_V + M_W$ and $S_F = B_{U,F} + B_{W,F} + B_{V,F}$.

- Assumptions 1 - 4 hold:

$$\mathbb{P}\left(\widetilde{z}_t \neq z_t\right) - \widehat{\mathcal{R}}_\gamma(f_t) \leq \widetilde{O}\left(\frac{tS_{2,1}\sum_{i=0}^{t-1} B_U^i}{\sqrt{m}\gamma}\right). \quad (1)$$

- Assumptions 1 - 3 and 5 hold:

$$\mathbb{P}\left(\widetilde{z}_t \neq z_t\right) - \widehat{\mathcal{R}}_\gamma(f_t) \leq \widetilde{O}\left(\frac{\Pi S_F \sum_{i=0}^{t-1} B_U^i \sqrt{d\ln(d)}}{\sqrt{m}\gamma}\right). \quad (2)$$

  - Bound (1) adapts the matrix covering lemma in [1].
  - Bound (2) adapts the PAC-Bayes approach in [3].

Comparison among Generalization Bounds

We compare different generalization bounds:

	Theorem 1	Bound (1)	Bound (2)
$B_U < 1$	$\widetilde{O}\left(\frac{d}{\sqrt{m}\gamma}\right)$	$\widetilde{O}\left(\frac{tS_{2,1}}{\sqrt{m}\gamma}\right)$	$\widetilde{O}\left(\frac{\sqrt{d}S_F}{\sqrt{m}\gamma}\right)$
$B_U = 1$	$\widetilde{O}\left(\frac{dt}{\sqrt{m}\gamma}\right)$	$\widetilde{O}\left(\frac{t^2 S_{2,1}}{\sqrt{m}\gamma}\right)$	$\widetilde{O}\left(\frac{dtS_F}{\sqrt{m}\gamma}\right)$
$B_U > 1$	$\widetilde{O}\left(\frac{\sqrt{d^3 t}}{\sqrt{m}\gamma}\right)$	$\widetilde{O}\left(\frac{tB_U^t S_{2,1}}{\sqrt{m}\gamma}\right)$	$\widetilde{O}\left(\frac{dB_U^t S_F}{\sqrt{m}\gamma}\right)$

Equivalent relation between matrix norms:

$$\|\cdot\|_2 \leq \|\cdot\|_{2,1} \leq \sqrt{d}\|\cdot\|_F \leq d\|\cdot\|_2$$

Compared to Theorem 1,

- Bound (2) is better, if $B_U < 1$.
- Bound (1) is better, if $tS_{2,1} < d$ and $B_U \leq 1$.
- Theorem 1 is better, if $B_U > 1$.

Proof Sketch

(I) PAC-learning Bound [2]

$$\mathbb{P}(\widetilde{z}_t \neq z_t) - \widehat{\mathcal{R}}_\gamma(f_t) \leq \mathfrak{R}_S(\mathcal{F}_{\gamma,t}) + 3\sqrt{\frac{\log \frac{2}{\delta}}{2m}}.$$

(II) Key Observation: Neural Networks are bi-Lipschitz. Consider $y = \sigma(Wx)$ with σ 1-Lipschitz.

- Given matrices W and W', we have
$$\|y - y'\|_2 = \|\sigma(Wx) - \sigma(W'x)\|_2 \leq \|x\|_2 \|W - W'\|_F.$$
- Given inputs x and x', we have
$$\|y - y'\|_2 = \|\sigma(Wx) - \sigma(Wx')\|_2 \leq \|W\|_2 \|x - x'\|_2.$$

Vanilla RNNs are multilayer networks.

Lemma 2. *Under Assumptions 1–3, given input $(x_t)_{t=1}^{T}$ and for any integer $t \leq T$, $\|y_t\|_2$ is Lipschitz in U, V and W, i.e.,*

$$\left\|y_t - y_t'\right\|_2 \leq L_{U,t}\left\|U - U'\right\|_F + L_{V,t}\left\|V - V'\right\|_F + L_{W,t}\left\|W - W'\right\|_F,$$

where $L_{U,t}, L_{V,t}$, and $L_{W,t}$ are coefficients.

Implication of Lemma 2:

- Coverings on weight matrices imply a covering on $\mathcal{F}_t$.

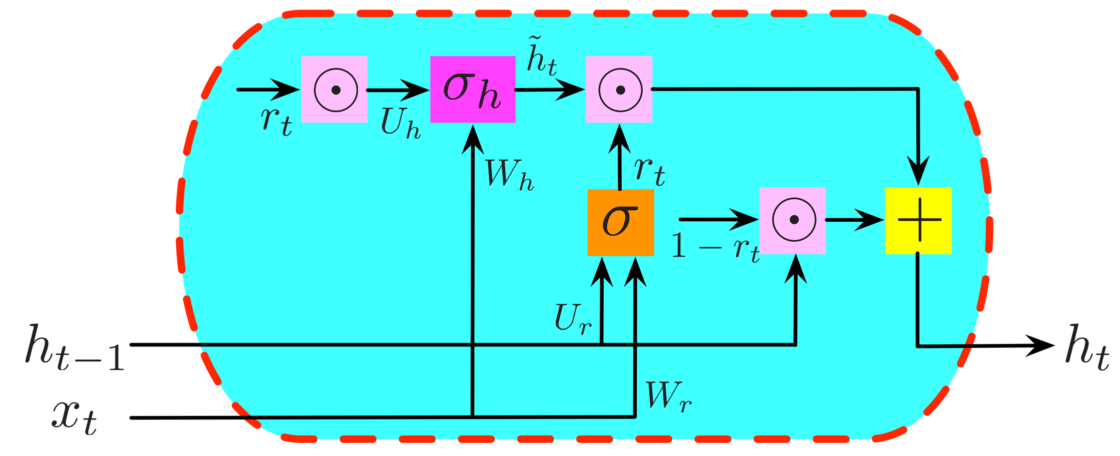
(III) Standard Machinery

- $\left.\begin{array}{l}\text{Volume ratio (separates } d\text{)} \\ \text{Covering under } \ell_{2,1} \text{ norm}\end{array}\right\} \overset{\text{bound}}{\Longrightarrow}$ Covering number.
- Covering number + Dudley's integral $\overset{\text{bound}}{\Longrightarrow} \mathfrak{R}_S(\mathcal{F}_{\gamma,t})$.

Extensions to MGU and LSTM

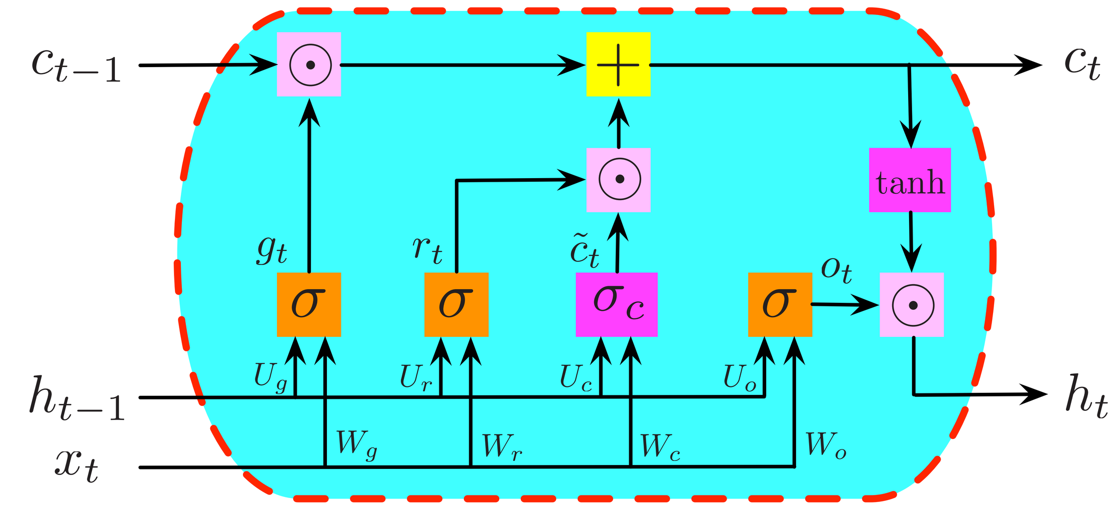
The MGU RNNs are the simplest gated RNNs, which take,

$$r_t = \sigma(W_r x_t + U_r h_{t-1}), \boxed{\widetilde{h}_t = \sigma_h\left(W_h x_t + U_h(r_t \odot h_{t-1})\right)},$$
$$h_t = (1 - r_t) \odot h_{t-1} + r_t \odot \widetilde{h}_t, \qquad y_t = \sigma_y(Vh_t).$$



The LSTM RNNs are more complicated, which take,

$$g_t = \sigma(W_g x_t + U_g h_{t-1}), \qquad r_t = \sigma(W_r x_t + U_r h_{t-1}),$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1}), \qquad \boxed{\widetilde{c}_t = \sigma_c\left(W_c x_t + U_c h_{t-1}\right)},$$
$$c_t = g_t \odot c_{t-1} + r_t \odot \widetilde{c}_t, \qquad h_t = o_t \odot \tanh(c_t).$$



MGU and LSTM introduce extra decaying factors on B_U.

- MGU: $B_U \Longrightarrow \|1 - r_t\|_\infty + B_{U_h}\|r_t\|_\infty^2$.
- LSTM: $B_U \Longrightarrow \|g_t\|_\infty + B_{U_c}\|r_t\|_\infty \|o_t\|_\infty$.

Under proper normalization, the generalization bounds of MGU and LSTM RNNs are **less dependent** on d and t.

> MGU and LSTM RNNs potentially **reduce** the dependence on d and t in generalization.

References

[1] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, pages 6241–6250, 2017.

[2] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2012.

[3] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

[4] J. Zhang, Q. Lei, and I. S. Dhillon. Stabilizing gradients for deep neural networks via efficient svd parameterization. *arXiv preprint arXiv:1803.09327*, 2018.