Combining Recurrent, Convolutional, and Continuous-time Models with Linear State-Space Layers

Докладчик: Василевская Юлия
Рецензент-исследователь: Боревский Андрей
Хакер: Пирогов Вячеслав

Что хотим делать? Какие подходы?

Хотим: обрабатывать данные, зависящие от времени (последовательности).

Подходы: CNN, RNN, NDE (neural differential equation)

Проблемы:

- ★ CNN: вычислительная сложность зависит от размера входа
- ★ RNN: сложно обрабатывать длинные последовательности (затухание/ взрыв градиентов)
- ★ NDE: неэффективны на практике

Встречаем Linear State-Space Layer (LSSL)

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (1)$$

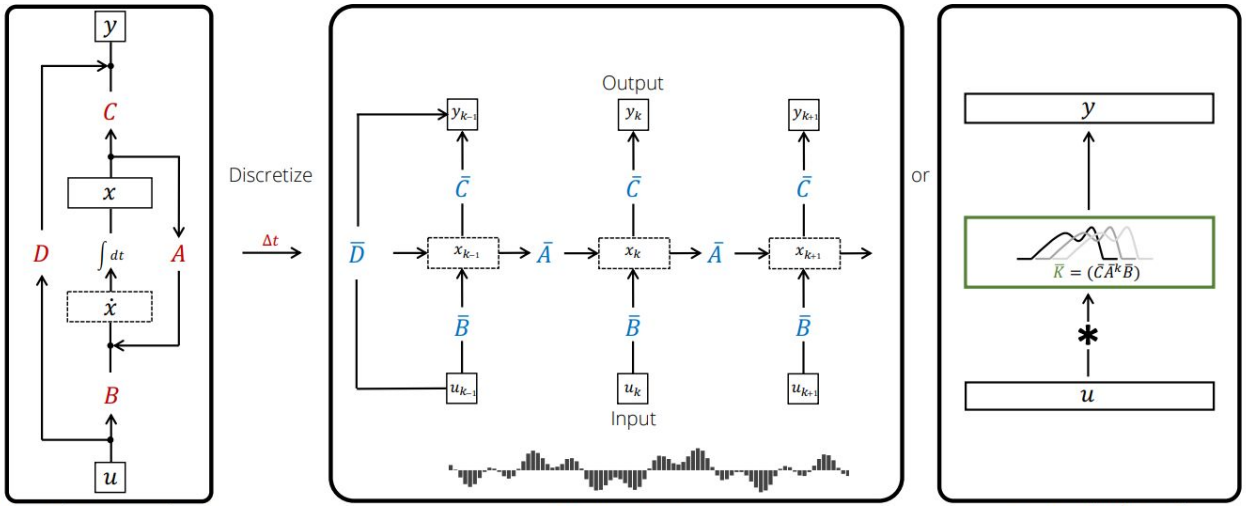
$$y(t) = Cx(t) + Du(t) \quad (2)$$

$u(t), y(t) \in \mathbb{R}$ - вход и выход модели, соответственно

$x(t) \in \mathbb{R}^N$ - скрытое состояние ($\dot{x}(t) \in \mathbb{R}^N$ - производная)

$A \in \mathbb{R}^{N \times N}, B \in \mathbb{R}^N, C \in \mathbb{R}^{1 \times N}, D \in \mathbb{R}$ - обучаемые параметры модели

Разные взгляды на LSSL



LSSL: связь с непрерывными моделями

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (1)$$
$$y(t) = Cx(t) + Du(t) \quad (2)$$

(1) из определения LSSL - линейное дифференциальное уравнение

Немного о дифференциальных уравнениях

$$\dot{x}(t) = f(t, x(t)) \Leftrightarrow x(t) = x(t_0) + \int_{t_0}^{t} f(s, x(s))ds$$

Тогда для дискретных входов t_0, t_1, t_2 ... имеет место

$$x(t_k) = x(t_{k-1}) + \int_{t_{k-1}}^{t_k} f(s, x(s))ds \Leftrightarrow x(t_k) - x(t_{k-1}) = \int_{t_{k-1}}^{t_k} f(s, x(s))ds$$

Generalized Bilinear Transform

Вспомним, что у нас имеет место быть: $f(t, x(t)) = Ax(t) + Bu(t)$

Возьмём: $u_k = \frac{1}{\Delta t_k} \int_{t_{k-1}}^{t_k} u(s)\, ds$ и получим:

$$x_k - x_{k-1} = \int_{t_{k-1}}^{t_k} (Ax(s) + Bu(s))ds = \int_{t_{k-1}}^{t_k} Ax(s)ds + \int_{t_{k-1}}^{t_k} Bu(s)ds = \int_{t_{k-1}}^{t_k} Ax(s)ds + \Delta t_k B u_k \approx$$
$$\approx \Delta t_k \left[(1-\alpha)Ax_{k-1} + \alpha Ax_k\right] + \Delta t_k B u_k \Leftrightarrow (I - \Delta t_k \alpha A)x_k = (I + \Delta t_k (1-\alpha)A)x_{k-1} + \Delta t_k B u_k$$

LSSL: связь с рекуррентными моделями

$$x_k = (I - \alpha\Delta t_k \cdot A)^{-1}(I + (1-\alpha)\Delta t_k \cdot A)x_{k-1} + (I - \alpha\Delta t_k \cdot A)^{-1}\Delta t_k \cdot Bu_k$$

И вот теперь мы видим рекуррентную зависимость скрытых состояний x_k. Осталось сделать замену и получить:

$$x_k = \overline{A}x_{k-1} + \overline{B}u_k \quad (3)$$

$$y_k = Cx_k + Du_k \quad (4)$$

LSSL: связь со свёрточными моделями

Из (3) и (4) при $x_{-1}=0$:

$$x_0 = \overline{A}x_{-1} + \overline{B}u_0 = \overline{B}u_0, \quad x_1 = \overline{A}x_0 + \overline{B}u_1 = \overline{AB}u_0 + \overline{B}u_1, \dots x_k = \overline{A}^k\overline{B}u_0 + \dots + \overline{AB}u_{k-1} + \overline{B}u_k$$

$$y_k = C\overline{A}^k\overline{B}u_0 + \dots + C\overline{AB}u_{k-1} + C\overline{B}u_k + Du_k$$

Вспомнив размерность $u_k \in \mathbb{R}, A \in \mathbb{R}^{N\times N}, B \in \mathbb{R}^N, C \in \mathbb{R}^{1\times N}, D \in \mathbb{R}$ осознаем, что перед нами 1-D свёрт $y = \mathcal{K}_L(\overline{A}, \overline{B}, C) * u + Du$ с

$$\mathcal{K}_L(A, B, C) = \left(CA^iB\right)_{i\in[L]} \in \mathbb{R}^L = (CB, CAB, \dots, CA^{L-1}B)$$

Проблемы LSSL

- ★ Обработка длинных последовательностей (наследуется от RNN)
- ★ Большая вычислительная сложность
  - ○ L матрично-векторных произведений при рекуррентном подходе
  - ○ Вычисление функции Крылова K_L при свёрточном подходе

Решение проблем: фиксированная матрица A

- ★ Случайная матрица A - плохо работает (с длинными последовательностями).
- ★ HiPPO - матрица работает хорошо.

HiPPO: High-Order Polynomial Projection Operator

$$A_{nk} = \begin{cases} (2n+1)^{1/2}(2k+1)^{1/2} & \text{if } n > k \\ n+1 & \text{if } n = k \\ 0 & \text{if } n < k \end{cases}$$

Решение проблем: обучаемая матрица A

Берём квазисепарабельную матрицу A и все проблемы решены :)

- ★ Матрично-векторное произведение: O(N) по времени
- ★ Функция Крылова: O(N + L) по памяти + квази-линейное время

В экспериментах берутся 3-квазисепарабельные матрицы.

LSSL в нейронных сетях

Есть: u_t и y_t - одномерные. Хотим: u_t и y_t - H-мерные.

Решение: обучаем H разных LSSL для каждой координаты.

Архитектура: LSSL + LayerNorm + SkipConnection + LSSL + LayerNorm + …

★ Можно из 1-D u_t получать M-D y_t, изменив размерности параметров A, B, C, D и dt (M - кол-во каналов в модели).

LSSL

- ★ Обучение: свёрточный вид
- ★ Применение: рекуррентный вид

LSSL-f: фиксированные A и dt

LSSL: обучаемые A и dt

Table 1: (**Pixel-by-pixel image classification.**) (Top) our methods. (Middle) recurrent baselines. (Bottom) convolutional + other baselines.

Model	sMNIST	pMNIST	sCIFAR
LSSL	**99.53**	**98.76**	**84.65**
LSSL-fixed	**99.50**	**98.60**	**81.97**
LipschitzRNN	99.4	96.3	64.2
LMUFFT [12]	-	98.49	-
UNIcoRNN [47]	-	98.4	-
HiPPO-RNN [24]	98.9	98.3	61.1
URGRU [25]	99.27	96.51	74.4
IndRNN [34]	99.0	96.0	-
Dilated RNN [8]	98.0	96.1	-
r-LSTM [56]	98.4	95.2	72.2
CKConv [44]	99.32	98.54	63.74
TrellisNet [4]	99.20	98.13	73.42
TCN [3]	99.0	97.2	-
Transformer [56]	98.9	97.9	62.2

Table 2: (**Vital signs prediction.**) RMSE for predicting respiratory rate (RR), heart rate (HR), and blood oxygen (SpO2). * indicates our own runs to complete results for the strongest baselines.

Model	RR	HR	SpO2
LSSL	**0.350**	**0.432**	**0.141**
LSSL-fixed	**0.378**	**0.561**	**0.221**
UnICORNN [47]	1.06	1.39	0.869*
coRNN [47]	1.45	1.81	-
CKConv	1.214*	2.05*	1.051*
NRDE [37]	1.49	2.97	1.29
IndRNN [47]	1.47	2.1	-
expRNN [47]	1.57	1.87	-
LSTM	2.28	10.7	-
Transformer	2.61*	12.2*	3.02*
XGBoost [55]	1.67	4.72	1.52
Random Forest [55]	1.85	5.69	1.74
Ridge Regress. [55]	3.86	17.3	4.16

Table 4: (**Raw Speech Classification; Timescale Shift.**) (Top): Raw signals (length 16000); $1 \to f$ indicates test-time change in sampling rate by a factor of f. (Bottom): Pre-processed MFCC features used in prior work (length 161). ✗ denotes computationally infeasible.

	LSSL	**LSSL-f**	CKConv	UnICORNN	N(C/R)DE	ODE-RNN [45]	GRU-ODE [16]
$1 \to 1$	**95.87**	90.64	71.66	11.02	16.49	✗	✗
$1 \to \frac{1}{2}$	**88.66**	78.01	65.96	11.07	15.12	✗	✗
MFCC	93.58	92.55	**95.3**	90.64	89.8	65.9	47.9

Table 3: (**Sequential CelebA Classification.**)

	LSSL-f	**ResNet**
Att.	78.89	81.35
MSO	92.36	93.92
Smil.	90.95	92.89
WL	90.57	93.25

Table 5: (**Modeling and Computational Benefits of LSSLs.**) In each benchmark category, we compare the number of epochs (ep.) it takes a LSSL-f to reach the previous SoTA (PSoTA) results as well as a near-SoTA target. We also report the wall clock time it took to reach PSoTA relative to the previous best model.

	Permuted MNIST			BDIMC Heart Rate			Speech Commands RAW		
	98% Acc.	PSoTA	Time	1.5 RMSE	PSoTA	Time	65% Acc.	PSoTA	Time
LSSL-fixed	16 ep.	104 ep.	0.19×	9 ep.	10 ep.	0.07×	9 ep.	10 ep.	0.14×
CKConv	118 ep.	200 ep.	1.0×	✗	✗	✗	188 ep.	280 ep.	1.0×
UnICORNN	75 ep.	✗	✗	116 ep.	467 ep.	1.0×	✗	✗	✗